

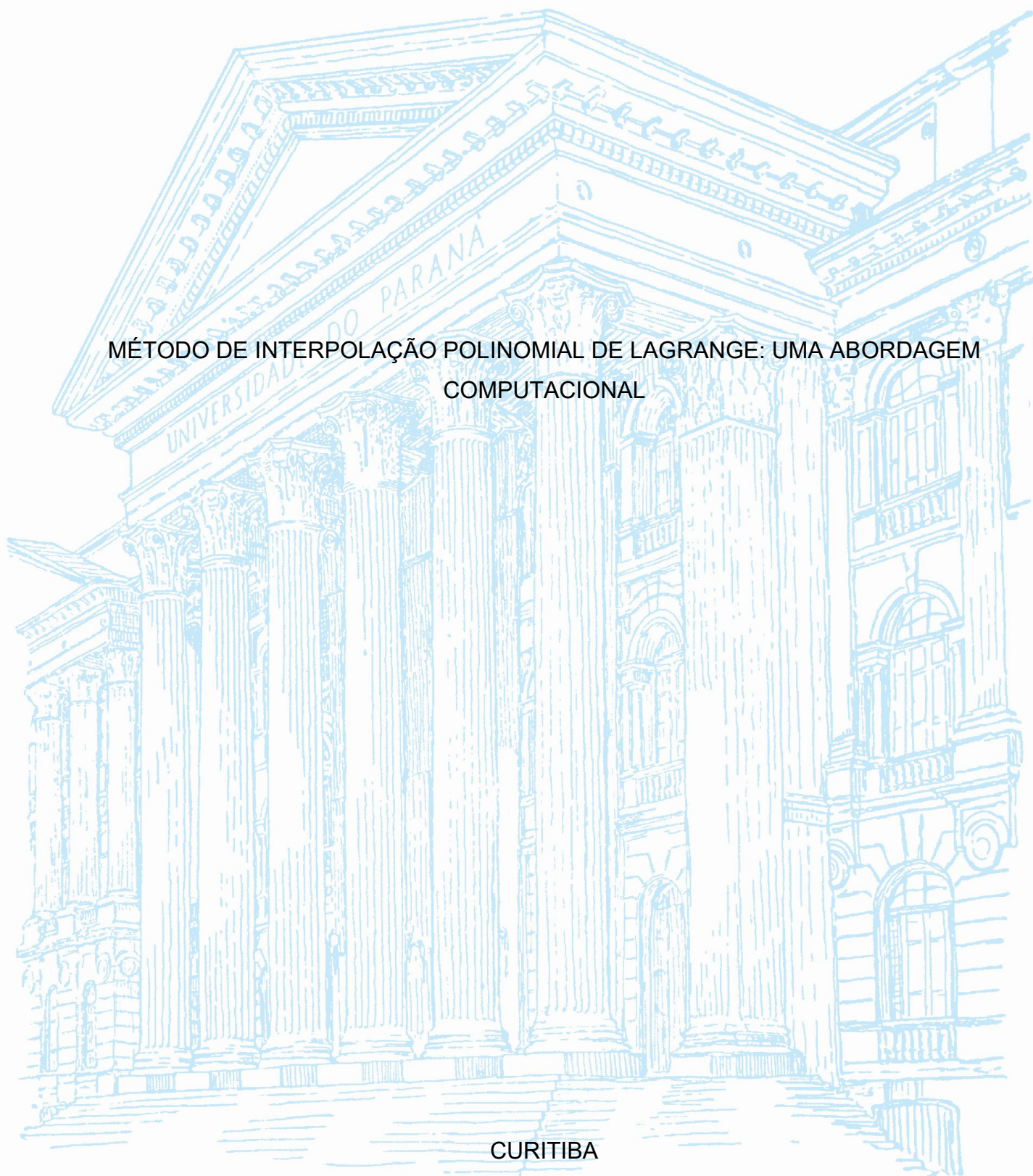
UNIVERSIDADE FEDERAL DO PARANÁ

PAULO GUSTAVO PAIXÃO BOTELHO

MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE: UMA ABORDAGEM
COMPUTACIONAL

CURITIBA

2022



PAULO GUSTAVO PAIXÃO BOTELHO

MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE: UMA ABORDAGEM
COMPUTACIONAL

Dissertação apresentada ao Programa de Mestrado Profissional em Matemática em Rede Nacional – PROFMAT, Setor de Ciências Exatas, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Mestre em Matemática.

Orientador: Prof. Dr. Roberto Pettres

CURITIBA

2022

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA JANDAIA DO SUL

Botelho, Paulo Gustavo Paixão

Método de interpolação polinomial de Lagrange : uma abordagem computacional. / Paulo Gustavo Paixão Botelho. – Curitiba, 2022.

1 recurso on-line : PDF.

Dissertação (Mestrado Profissional) – Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Matemática em Rede Nacional - PROFMAT.

Orientador: Prof. Dr. Roberto Pettres.

1. Interpolação polinomial de Lagrange . 2. Relações de Girard.
3. Planilha eletrônica de cálculo. 4. Método de Newton. 5. Métodos computacionais em C#. I. Pettres, Roberto. II. Universidade Federal do Paraná. III. Título.

CDD: 515.55

ATA DE SESSÃO PÚBLICA DE DEFESA DE MESTRADO PARA A OBTENÇÃO DO GRAU DE MESTRE EM MATEMÁTICA

No dia vinte e nove de abril de dois mil e vinte e dois às 19:30 horas, na sala Ambiente Virtual para defesas de dissertação PROFORMAT. <https://teams.microsoft.com/join/19%3ad668ba8eb35b4bcd80afb0c8dc23bfe7%40thread.tacv2/1648475319125?context=%7b%22Tid%22%3a%22c37b37a3-e9e2-42f9-bc67-4b9b738e1df0%22%2c%22Oid%22%3a%223689b35c-7a62-40d1-a735-b7faa14a5be9%22%7d>, Ambiente Virtual para defesas de dissertação PROFORMAT. <https://teams.microsoft.com/join/19%3ad668ba8eb35b4bcd80afb0c8dc23bfe7%40thread.tacv2/1648475319125?context=%7b%22Tid%22%3a%22c37b37a3-e9e2-42f9-bc67-4b9b738e1df0%22%2c%22Oid%22%3a%223689b35c-7a62-40d1-a735-b7faa14a5be9%22%7d>, foram instaladas as atividades pertinentes ao rito de defesa de dissertação do mestrando **PAULO GUSTAVO PAIXÃO BOTELHO**, intitulada: **MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE: UMA ABORDAGEM COMPUTACIONAL**, sob orientação do Prof. Dr. ROBERTO PETTRES. A Banca Examinadora, designada pelo Colegiado do Programa de Pós-Graduação MATEMÁTICA EM REDE NACIONAL da Universidade Federal do Paraná, foi constituída pelos seguintes Membros: ROBERTO PETTRES (UNIVERSIDADE FEDERAL DO PARANÁ), GERSON ULBRICHT (INSTITUTO FED. DE EDUC., CIÊNC. E TECNOL. DE SANTA CATARINA), ELIANDRO RODRIGUES CIRILO (40002012). A presidência iniciou os ritos definidos pelo Colegiado do Programa e, após exarados os pareceres dos membros do comitê examinador e da respectiva contra argumentação, ocorreu a leitura do parecer final da banca examinadora, que decidiu pela APROVAÇÃO. Este resultado deverá ser homologado pelo Colegiado do programa, mediante o atendimento de todas as indicações e correções solicitadas pela banca dentro dos prazos regimentais definidos pelo programa. A outorga de título de mestre está condicionada ao atendimento de todos os requisitos e prazos determinados no regimento do Programa de Pós-Graduação. Nada mais havendo a tratar a presidência deu por encerrada a sessão, da qual eu, ROBERTO PETTRES, lavrei a presente ata, que vai assinada por mim e pelos demais membros da Comissão Examinadora.

Assinatura Eletrônica

29/04/2022 22:27:27.0

ROBERTO PETTRES

Presidente da Banca Examinadora

Assinatura Eletrônica

03/05/2022 15:33:06.0

GERSON ULBRICHT

Avaliador Externo (INSTITUTO FED. DE EDUC., CIÊNC. E TECNOL. DE SANTA CATARINA)

Assinatura Eletrônica

04/05/2022 06:53:38.0

ELIANDRO RODRIGUES CIRILO

Avaliador Externo (40002012)



TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação MATEMÁTICA EM REDE NACIONAL da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **PAULO GUSTAVO PAIXÃO BOTELHO** intitulada: **MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE: UMA ABORDAGEM COMPUTACIONAL**, sob orientação do Prof. Dr. ROBERTO PETTRES, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 29 de Abril de 2022.

Assinatura Eletrônica

29/04/2022 22:27:27.0

ROBERTO PETTRES

Presidente da Banca Examinadora

Assinatura Eletrônica

03/05/2022 15:33:06.0

GERSON ULBRICHT

Avaliador Externo (INSTITUTO FED. DE EDUC., CIÊNC. E TECNOL. DE SANTA CATARINA)

Assinatura Eletrônica

04/05/2022 06:53:38.0

ELIANDRO RODRIGUES CIRILO

Avaliador Externo (40002012)

AGRADECIMENTOS

Ao Grande Arquiteto Do Universo, que é Deus, fonte fecunda de luz, de felicidade e de virtude, por tudo.

À minha Mãe Sônia que me deu à Luz, que me fez quem sou - sim, é a culpada - e de quem herdei o gosto pelos números, pelas coisas exatas e pela exatidão das coisas.

Ao meu Filho Luís Paulo, a quem amo mais que tudo e de quem tenho um orgulho que não cabe no peito.

À minha Esposa Andréa que, apesar de ter demorado muito, finalmente veio fazer parte da minha vida, por todo o carinho e amor dedicados a mim e aos meus.

À minha Avó Cleonice pelo exemplo, pela ternura sem limites e por estar até hoje conosco nos ensinando sempre.

Ao meu Avô Rodolpho Gustavo, cuja ausência sinto até o fundo de minh'alma, por tudo o que representou na minha vida.

Ao meu Orientador, Professor Doutor Roberto Pettres, por toda a paciência e ensinamentos.

Sem os acima citados, cada qual com sua parcela e significância, jamais teria chegado aqui.

“A matemática é a linguagem com a qual Deus escreveu o Universo“. Galileu Galilei

“Se vi mais longe, foi porque estava sobre os ombros de gigantes” Sir Isaac Newton

RESUMO

Apresentamos como o Método de Interpolação Polinomial de (Joseph Louis) Lagrange é usado para, a partir de um conjunto de pontos conhecidos, calcular a ordenada de outros pontos que sejam aderentes ao padrão de comportamento apresentado por esses pontos conhecidos ou determinar o polinômio que melhor represente esse padrão. Com o intuito de estimular o uso dos recursos computacionais hoje largamente disponíveis para todos, e aguçar a curiosidade do leitor na aplicação da matemática para solucionar problemas do mundo real, apresentamos também formas de aplicação deste Método através da montagem de planilhas de cálculo eletrônicas e do desenvolvimento de programas de computador, utilizando a linguagem C#. Apresentamos, ainda, as Relações de Girard para ajudar na determinação do polinômio interpolador e citamos, de forma breve, como o Método de Newton para determinar raízes de polinômios pode ser adaptado para resolver o problema inverso, qual seja, obter a abscissa de outros pontos a partir de suas ordenadas.

Palavras-chave: Interpolação Polinomial de Lagrange, Relações de Girard, Planilha Eletrônica de Cálculo, Método de Newton, Métodos Computacionais em C#.

ABSTRACT

We present how the Lagrange's Polynomial Interpolation Method is applied to, from a set of known points, find out the ordinate of other points that are adherent to the pattern defined by the known points or determine the polynomial expression that better represents this pattern. In order to estimate the use of the computational resources that are widely available nowadays for all people and pique the reader curiosity on how to use the mathematic to solve real problems, we also present how to apply this Method by using electronic calculation spreadsheets and by developing computational programs using C# language. In addition, we present the Girard's Relations that help us to calculate the interpolating polynomial and we mention how the Newton's Method, used to calculate the roots of polynomial expressions, can be adapted to solve the reverse scenario, that is, to calculate the abscissa of other points given its ordinates.

Keywords: Lagrange's Polynomial Interpolation, Girard's Relations, Electronic Calculation Spreadsheets, Newton's Method, Computational Methods in C#.

LISTA DE ILUSTRAÇÕES

Figura 1 - Exemplo de gráfico usando o método da Interpolação Linear	15
Figura 2 - Exemplo de gráfico usando o método da Interpolação Polinomial.....	16
Figura 3 - Exemplo de 3 pontos distintos que, por serem colineares, serão interpolados através de uma reta e não de uma parábola	34
Figura 4 - Parábola com os três pontos distintos A, B e C conhecidos (em azul) e o ponto P a descobrir (em vermelho)	36
Figura 5 - Armazenamento das Coordenadas dos DOIS Pontos Conhecidos	40
Figura 6 - Diferenças entre as abscissas dos pontos conhecidos, duas a duas, calculadas em células da Matriz de Valores de X	41
Figura 7 - Células para as Coordenadas do Ponto Desconhecido.....	41
Figura 8 - Vetor Linha onde serão calculados $L_0(x)$ e $L_1(x)$	42
Figura 9 - Célula onde aparecerá o Valor da Ordenada do Ponto Desconhecido	42
Figura 10 - Teste 1 - Usando a Abscissa do Primeiro Ponto Conhecido	43
Figura 11 - Teste 2 - Usando a Abscissa do Segundo Ponto Conhecido	44
Figura 12 - Teste 3 - Usando a Abscissa do Ponto Desconhecido.....	45
Figura 13 - Armazenamento das Coordenadas dos TRÊS Pontos Conhecidos	45
Figura 14 - Vetor Linha ampliado para conter $L_2(x)$	46
Figura 15 - Teste 1 - Usando a Abscissa do Primeiro Ponto Conhecido	46
Figura 16 - Teste 2 - Usando a Abscissa do Segundo Ponto Conhecido	47
Figura 17 - Teste 3 - Usando a Abscissa do Terceiro Ponto Conhecido.....	48
Figura 18 - Teste 4 - Usando a Abscissa do Ponto Desconhecido.....	48
Figura 19 - Algoritmo em Linguagem Natural para Implementação da Forma Direta .	52
Figura 20 - Vetores das Coordenadas X e Y já preenchidos com os DOIS pontos conhecidos	54
Figura 21 - Código-Fonte do Programa de Implementação da Forma Direta.....	54
Figura 22 - O programa pede que o usuário informe a abscissa do ponto cuja ordenada deve ser calculada.....	55
Figura 23 - Resultado do Teste 1 - Primeiro Ponto Conhecido	55
Figura 24 - Resultado do Teste 2 - Segundo Ponto Conhecido	55
Figura 25 - Resultado do Teste 3 - Ponto Desconhecido.....	55
Figura 26 - Vetores das Coordenadas X e Y já preenchidos com os TRÊS pontos conhecidos	56
Figura 27 - Resultado do Teste 1 - Primeiro Ponto Conhecido	56
Figura 28 - Resultado do Teste 2 - Segundo Ponto Conhecido	56
Figura 29 - Resultado do Teste 3 - Terceiro Ponto Conhecido.....	57
Figura 30 - Resultado do Teste 4 - Ponto Desconhecido.....	57
Figura 31 - Armazenamento das Coordenadas dos DOIS Pontos Conhecidos.....	61

Figura 32 - Inclusão de Quadro para Cálculos de D_n e (y_n / D_n)	61
Figura 33 - Inclusão de Quadro para Cálculos dos Coeficientes de N_n	62
Figura 34 - Inclusão de Quadro para Multiplicar os Coeficientes por (y_n / D_n)	62
Figura 35 - Linha de Somatório que Traz os Coeficientes Finais de $p_n(x)$	62
Figura 36 - Planilha Completa para Aplicação da Forma Indireta para DOIS Pontos Conhecidos	64
Figura 37 - Planilha com Uma Linha a Mais. O denominador D passa a ter DOIS fatores	65
Figura 38 - Matrizes de Coeficientes Modificadas para mais um Ponto Conhecido e mais um Grau no Polinômio Interpolador	66
Figura 39 - Teste 1 - Usando a Abscissa do Primeiro Ponto Conhecido	66
Figura 40 - Teste 2 - Usando a Abscissa do Segundo Ponto Conhecido.....	66
Figura 41 - Teste 3 - Usando a Abscissa do Terceiro Ponto Conhecido.....	66
Figura 42 - Teste 4 - Usando a Abscissa do Ponto Desconhecido.....	66
Figura 43 - Planilha Completa para Aplicação da Forma Indireta para TRÊS Pontos Conhecidos	67
Figura 44 - Teste com TRÊS Pontos Conhecidos, porém Colineares	68
Figura 45 - Algoritmo em Linguagem Natural para Implementação da Forma Indireta	72
Figura 46 - Parte Inicial do Programa - Definição das Variáveis - Vetores de Coordenadas preenchidos com as Coordenadas dos DOIS Pontos Conhecidos	73
Figura 47 - Parte Intermediária do Programa - Lógica para Cálculo dos D_n	74
Figura 48 - Parte Final do Programa - Lógica para Cálculo dos Coeficientes Parciais e Finais e Exposição dos Resultados em Tela.....	75
Figura 49 - Resultado do Teste 1 - Primeiro Ponto Conhecido	75
Figura 50 - Resultado do Teste 2 - Segundo Ponto Conhecido.....	76
Figura 51 - Resultado do Teste 3 - Ponto Desconhecido.....	76
Figura 52 - Parte Inicial do Programa - Definição das Variáveis - Vetores de Coordenadas preenchidos com as Coordenadas dos TRÊS Pontos Conhecidos	77
Figura 53 - Resultado do Teste 1 - Primeiro Ponto Conhecido	77
Figura 54 - Resultado do Teste 2 - Segundo Ponto Conhecido	77
Figura 55 - Resultado do Teste 3 - Terceiro Ponto Conhecido.....	78
Figura 56 - Resultado do Teste 4 - Ponto Desconhecido.....	78
Figura 57 - Vetores de Coordenadas Preenchidos com TRÊS Pontos Colineares.....	78
Figura 58 - Teste com TRÊS Pontos Conhecidos, porém Colineares	79
Figura 59 - $t(x)=x^4 - 2; 33x^3 - 6; 25x^2 + 3; 5x + 1; 5$ plotada via GeoGebra	80
Figura 60 - Vetores de Coordenadas preenchido com os CINCO Pontos Conhecidos	80
Figura 61 - Resultado do Teste I de Validação Final	81

Figura 62 - Vetores de Coordenadas preenchido com SEIS Pontos Conhecidos . . .	81
Figura 63 – Resultado do Teste II de Validação Final	81
Figura 64 – Translação Vertical para uso do Método de Newton na Resolução do Problema Inverso	86

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Generalidades.....	14
1.2	Justificativa	16
1.3	Objetivos	16
1.3.1	Objetivo Geral	16
1.3.2	Objetivos Específicos	16
1.4	Organização do Trabalho	17
2	REVISÃO DE LITERATURA.....	19
2.1	Quem foi Lagrange	19
2.2	O Método de Interpolação Polinomial de Lagrange	20
2.2.1	Consideração Iniciais	20
2.3	O Desenvolvimento do Método	24
2.4	Estudo da Unicidade do Polinômio Interpolador.....	27
3	DESENVOLVIMENTO ALGÉBRICO DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE	29
3.1	O Polinômio Interpolador para um Único Ponto Conhecido.....	29
3.2	O Polinômio Interpolador para Dois Pontos Distintos Conhecidos .	29
3.3	O Polinômio Interpolador para Três Pontos Distintos Conhecidos	31
3.4	Estudo do Grau do Polinômio Interpolador	33
4	DESENVOLVIMENTO NUMÉRICO DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE	35
4.1	Forma Direta para Cálculo da Imagem de um Ponto Não Conhecido	35
4.2	Forma Indireta para Cálculo da Imagem de um Ponto Não Conhecido	36
4.3	Retorno ao Estudo do Grau do Polinômio Interpolador	39
5	DESENVOLVIMENTO COMPUTACIONAL DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE - FORMA DIRETA	40
5.1	Cálculo pela Forma Direta Utilizando uma Planilha Eletrônica.....	40
5.2	A Fórmula Geral Escrita em Formato Iterativo	49
5.3	O Algoritmo que Implementa os Laços de Repetição	51
5.4	O Programa que Implementa os Laços de Repetição.....	53
6	DESENVOLVIMENTO COMPUTACIONAL DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE - FORMA INDIRETA .	58

6.1	Determinação de uma Fórmula para Cálculo dos Coeficientes do Polinômio Interpolador.....	58
6.2	Cálculo pela Forma Indireta Utilizando uma Planilha Eletrônica.....	60
6.3	A Fórmula Geral Escrita em Formato Iterativo	68
6.4	O Algoritmo que Implementa os Laços de Repetição	68
6.5	O Programa que Implementa os Laços de Repetição.....	72
6.6	Validação Final	79
7	ESTUDO ADICIONAL - O PROBLEMA INVERSO: CALCULAR A ABSCISSA A PARTIR DA ORDENADA	83
7.1	Método Aproximativo com uma Vizinhança Definida	83
7.2	Método Aproximativo com uma Vizinhança Definida	85
7.3	O Método de Newton	85
8	CONSIDERAÇÕES FINAIS	87
	Referências.....	89

1 INTRODUÇÃO

1.1 Generalidades

Durante o estudo regular da matemática, e mais particularmente quando estuda funções, o estudante se depara, na grande maioria das vezes, com situações em que é conhecida a função em termos da expressão $y = f(x)$ ¹. Essa expressão estabelece uma correlação entre os valores que a variável x , chamada de variável independente, pode assumir (respeitadas as regras de restrição de domínio, quando existentes) e os respectivos valores resultantes para a variável y , chamada de variável dependente².

Considerando que geralmente lida-se com funções de $\mathbb{R} \rightarrow \mathbb{R}$, ou seja, aquelas em que a variável independente assume valores reais que resultam em valores também reais para a variável dependente, o que equivale a dizer que o domínio da função (D_f) é o conjunto dos números reais (novamente enfatizando a necessidade de se respeitar as regras de restrição de domínio, quando existentes) e o contra-domínio (CD_f) também é o conjunto dos números reais, há uma infinidade de valores possíveis para x e, por conseguinte, também uma infinidade de valores para y (exceto nos casos particulares das funções constantes).

Usando-se a já previamente conhecida expressão da função na forma $y = f(x)$, esta pode ser encarada como uma espécie de “receita de bolo”, que estabelece os passos a serem aplicados sobre o “ingrediente” (variável x) para produzir (calcular) o valor da variável y . O fato de o ponto de partida ser um valor (válido, pertencente ao domínio) atribuído a x para se chegar ao respectivo valor de y é que faz com que a variável x seja chamada de independente e a variável y seja chamada de dependente.

Essa correspondência entre as variáveis independente e dependente, x e y , cria infinitos pares ordenados representados como (x, y) e que identificam infinitos pontos dentro de um plano cartesiano do \mathbb{R}^2 que, uma vez posicionados no plano e ligados, produzirão o gráfico da função.

Entretanto, é muito comum, especialmente na área da engenharia, que através de experimentações ou amostragem seja obtido um conjunto de pontos discretos e **não contínuos** que representam o comportamento de uma função baseado nesse número limitado de pontos. A partir dessa amostra de pontos, deseja-se conhecer os valores que esta função assumirá em pontos intermediários não mapeados. Essas são justamente situações que representam um problema inverso do discutido até aqui, qual seja, há alguns (ou muitos) pontos, pares ordenados (x, y) a partir dos quais deseja-se estabelecer uma

¹ Expressão genérica para os casos de funções de uma variável independente. As afirmações são válidas também para as funções que possuem mais de uma variável independente.

² Cabe ressaltar que não serão feitas aqui considerações acerca das regras que estabelecem quando uma relação é, de fato, uma função; tampouco serão feitas considerações acerca das regras que definem se esta correspondência entre a variável independente e a variável dependente é injetora, sobrejetora ou mesmo bijetora.

função $y = f(x)$ que satisfaça esses pontos, ou seja, cujo gráfico contenha todos estes pontos.

A esse processo, ou método, de encontrar a expressão de uma função a partir de um conhecido conjunto de pontos, dá-se o nome de **interpolação**.

Existem diversos métodos de interpolação.

Um dos métodos de interpolação bastante conhecido é o de **interpolação linear**. O leitor mais atento poderá, neste momento, questionar se uma interpolação linear é apropriada uma vez que nada garante que os pontos conhecidos sejam colineares. Se o forem, o problema é de fácil resolução, bastando tomar dois pontos quaisquer da coleção de pontos conhecidos e calcular a expressão da reta que passará por aqueles dois pontos escolhidos e, por conseguinte, por todos os demais pontos da coleção, uma vez que são todos colineares.

Entretanto, no caso em que os pontos conhecidos não são colineares, a interpolação linear consiste em determinar a expressão das retas que ligam dois pontos adjacentes, fazendo com que o valor que a função assumirá nos pontos intermediários seja calculado com base na reta que une os dois pontos adjacentes conhecidos e que serviram de extremidade para a determinação dessa reta.

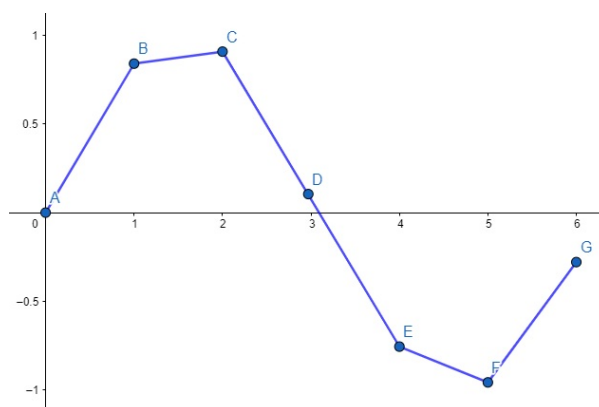


Figura 1 – Exemplo de gráfico usando o método da Interpolação Linear

Como uma reta também é expressa através de um polinômio, uma interpolação linear gerará uma função polinomial que será expressa em partes, uma vez que para cada reta unindo dois pontos adjacentes haverá uma reta diferente. Já a interpolação polinomial é assim chamada porque produz como função interpoladora um polinômio de expressão única.

A Figura 2 a seguir mostra os mesmos sete pontos da Figura 1, porém interpolados através de um polinômio, e não mais por retas.

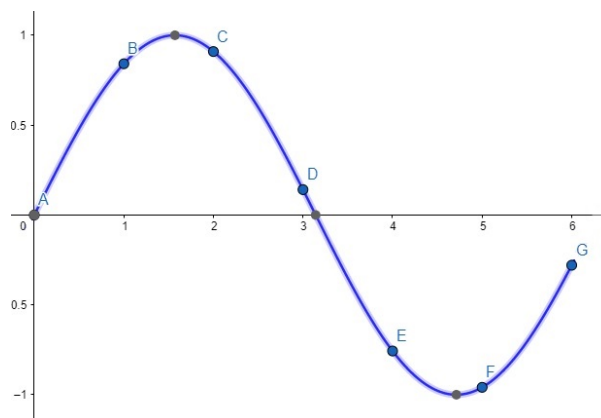


Figura 2 – Exemplo de gráfico usando o método da Interpolação Polinomial.

Existem vários métodos de Interpolação Polinomial (Taylor, Hermite, Newton) mas este trabalho tem por objetivo apresentar o Método de Interpolação Polinomial desenvolvido pelo matemático ítalo-francês Joseph Louis **Lagrange**, suas vantagens e desvantagens e, a partir dele, iniciar e motivar o estudante à construção de rotinas em forma de algoritmos que realizam operações matemáticas sequencialmente como se fossem um programa de computador. Ainda, serão apresentados os programas que implementam tais algoritmos.

1.2 Justificativa

Apresentar o Método de Interpolação Polinomial de Lagrange como forma de aproximar através de uma função um conjunto de pares ordenados proveniente de uma tabela ou pesquisa estatística, e incentivar o leitor, principalmente professores de matemática e estudantes do Ensino Médio, para o uso de recursos computacionais para aplicação de métodos matemáticos.

1.3 Objetivos

1.3.1 Objetivo Geral

Desenvolver uma abordagem prática que pode servir como material didático de apoio aos docentes de Matemática quando tratarem de Funções de Uma Variável, ou mesmo os de Matemática Computacional, incluindo a construção de algoritmos (e o seu desenvolvimento em linguagem de programação) interconectando a teoria com a prática.

1.3.2 Objetivos Específicos

Para organizar os passos a serem percorridos com a finalidade de atingir o Objetivo Geral deste trabalho, foram definidos os seguintes Objetivos Específicos:

- 1) Apresentar uma rápida contextualização histórica sobre Lagrange e o seu importante papel no desenvolvimento da matemática na sua época;
- 2) Apresentar o Método de Interpolação Polinomial de Lagrange, apontando cenários nos quais o método é (ou não é) aplicável, desenvolvendo cenários específicos para uma quantidade fixa de pontos conhecidos;
- 3) Desenvolver uma Fórmula Geral do Método para uma quantidade indefinida de pontos conhecidos (fundamental para a construção de um algoritmo genérico);
- 4) Desenvolver métodos computacionais usando planilha eletrônica, para cálculo da imagem $f(x)$ de um ponto não conhecido a partir de sua abscissa utilizando a Fórmula Geral do Método e para determinação do Polinômio Interpolador;
- 5) Construir o algoritmo que calculará esta imagem (que é um dos objetivos finais do método em si) utilizando a Fórmula Geral do Método e o que calculará o polinômio (e não a imagem de um ponto dado), a partir da Fórmula Geral do Método convenientemente adaptada;;
- 6) Programar estes algoritmos em linguagem de programação (usaremos C#)
- 7) Mostrar, como estudo adicional sugerido, como o Polinômio Interpolador pode ser usado para a resolução do problema inverso: dada uma imagem $f(x)$, calcular qual o valor de x correspondente a ela.

1.4 Organização do Trabalho

Este trabalho está dividido em **oito** capítulos.

Neste Capítulo 1 são apresentados o problema a ser estudado, os objetivos gerais e específicos a serem atingidos, e os passos que serão percorridos para atingi-los.

No Capítulo 2 é apresentado ao leitor um brevíssimo relato sobre quem foi Joseph Louis Lagrange e sua importância para o desenvolvimento da matemática no século XVIII.

No Capítulo 3 é apresentado o Método propriamente dito, sua aplicabilidade e restrições, além de seu desenvolvimento algébrico para 2, 3 e n pontos conhecidos, com a análise de algumas características do polinômio interpolador, inclusive sua unicidade e seu grau.

No Capítulo 4 são apresentadas duas Formas – Direta e Indireta – de utilização do Método com o auxílio de dois exemplos numéricos que também serão usados nos capítulos seguintes.

No Capítulo 5 é melhor detalhada a Forma Direta de utilização do Método, onde é construída uma Planilha Eletrônica para sua aplicação, é escrito um Algoritmo em linguagem natural (português não-estruturado) e é desenvolvido um Programa de Computador em

linguagem C#, usando-se os dois exemplos numéricos apresentados no capítulo anterior para validar essas soluções desenvolvidas.

No Capítulo 6 é feito o mesmo desenvolvimento que foi feito no capítulo 5, mas dessa vez para a Forma Indireta, também fazendo uso de Planilha Eletrônica, Algoritmo e Programa em C#.

No Capítulo 7, como sugestão de estudo adicional, é mostrada a aplicação do polinômio interpolador calculado pela Forma Indireta para a resolução do problema inverso: calcular x a partir de uma imagem $f(x)$ dada.

No Capítulo 8, estão as Considerações Finais.

2 REVISÃO DE LITERATURA

2.1 Quem foi Lagrange

Nascido em 25 de janeiro de 1736, na cidade italiana de Turim, Joseph Louis Lagrange [1](batizado de Giuseppe Luigi Lagrangia), ou simplesmente Lagrange, foi o mais jovem de onze filhos, tendo sido o único que não morreu na infância.

Segundo BOYER [2], Lagrange é geralmente considerado como “o mais notável matemático do século XVIII”, comparável apenas com Euler³, por conta das importantes contribuições que fez em diversas áreas, com destaque para o Cálculo Variacional que resultou na Equação de Euler-Lagrange.

Mudou-se de Turim para Berlim em 1766 a convite do Rei Frederico da Prússia, permanecendo em sua corte até a morte do monarca ocorrida em 1786, quando, então, se mudou para Paris, a convite de Luís XVI, onde integrou a Academia Francesa de Ciências. Apesar da Revolução Francesa de 1789 ter deposto e guilhotinado diversos membros da aristocracia francesa, Lagrange não sofreu qualquer ameaça dos revolucionários, tendo, inclusive, recebido honras e distinções até mesmo do Imperador Napoleão Bonaparte anos depois. Em 1799 foi nomeado senador e, em 1802, adquiriu cidadania francesa após a sua terra natal, o Piemonte, ter sido anexada à França.

Lecionou na Escola Politécnica a partir de 1794 e participou das discussões que culminaram na adoção das unidades metro e quilograma pela França, em 1799.

Faleceu em Paris, em 10 de abril de 1813, somente dois dias após receber das mãos de Napoleão a Grande Cruz da Ordem Imperial da Reunião.⁴

Ainda, Lagrange foi:

- Eleito membro da Academia de Berlim em 02 de setembro de 1756;
- Eleito membro da Sociedade Real de Edimburgo em 1790;
- Membro da Sociedade Real britânica;
- Eleito Membro da Real Academia de Ciências da Suécia em 1806;
- Feito Grande Oficial da Legião de Honra por Napoleão em 1808
- Feito Conde do Império por Napoleão em 1808.

³ Leonhard Euler (1707-1783), nascido na Basileia, Suíça

⁴ Para saber sobre a Ordem Imperial da Reunião: https://en.wikipedia.org/wiki/Order_of_the_Reunion

2.2 O Método de Interpolação Polinomial de Lagrange

2.2.1 Consideração Iniciais

O Método de Interpolação Polinomial de Lagrange, é um assunto tratado em diversos outros trabalhos dentro e fora do programa do PROFMAT. Como exemplos, destacam-se:

GABETTA JÚNIOR [3], apresenta um estudo bastante amplo e interessante sobre a aproximação de funções por interpolação. Após definir os conceitos de Função, Polinômio e Função Polinomial, ele apresenta a Interpolação Polinomial em seu capítulo 3 e, mais especificamente na seção 3.2.3, apresenta o Método de Interpolação Polinomial de Lagrange, demonstrando como se chega à Fórmula Geral e aplicando alguns exemplos numéricos para demonstrar sua utilidade. É importante destacar a seção 3.2.4 onde o autor cita o Fenômeno de Runge associado à ideia, equivocada, de que quanto maior for a quantidade de pontos melhor será a função interpolante, e as seções 3.2.5 e 3.2.6 onde são feitos comentários sobre o estudo dos erros que podem surgir ao se fazer a aproximação de uma função conhecida através da determinação de uma função interpolante. Este tema, inclusive, está em destaque no Anexo daquele trabalho onde o autor explora o Teorema do Erro da Interpolação Polinomial a partir do Teorema de Rolle. O principal objetivo de GABETTA em seu trabalho é tratar o Método de Interpolação Polinomial de Lagrange sob uma perspectiva teórica algébrica e apresentar o software GeoGebra como um recurso computacional a ser utilizado para uma abordagem geométrica do Método.

SILVA [4], apresenta em seu trabalho o Método de Interpolação Polinomial de Lagrange, demonstrando como se chega à Fórmula Geral e apresenta um exemplo numérico aplicando o GeoGebra para plotar o Polinômio Interpolador e permitir sua compreensão geométrica. Finalmente, em seu capítulo 5, o autor apresenta sugestões de estratégias para a utilização deste Método no ensino de modelagem matemática no Ensino Médio.

MARTINS [5], também faz uso do Método de Interpolação Polinomial de Lagrange para abordar o tema da modelagem matemática, buscando uma aproximação entre teoria e prática numa tentativa de tornar a matemática mais familiar ao alunos e facilitar sua aprendizagem.

SILVA [6], apresenta três métodos para calcular o polinômio interpolador: um sistema linear com tantas equações quantas forem os pontos distintos conhecidos, o Método de Lagrange - apontado por ele como sendo mais eficiente do que a resolução do sistema de equações - e ainda o Método de Newton de Diferenças Divididas. No Capítulo 4 o autor faz algumas considerações interessantes sobre o estudo dos Erros na Interpolação e finalmente, no Capítulo 5, apresenta um proposta didática para o ensino da interpolação polinomial em uma turma do Ensino Médio com utilização do GeoGebra.

AMES [7] em seu interessante trabalho também apresenta os Métodos de Sistemas

Lineares, de Lagrange e de Newton para determinação do polinômio interpolador, mas também os Métodos dos Polinômios de Chebyshev e o Método dos Polinômios de Legendre. A autora afirma que "... a Forma de Newton é utilizada para a análise de erro; a Forma de Lagrange é usada para dedução de cálculo aproximado de área (integral); a Forma de Legendre é usada para fórmulas de quadratura e os Polinômios de Chebyshev para integração numérica".

Vários outros trabalhos que tratam do Método de Interpolação Polinomial de Lagrange podem ser encontrados na homepage do PROFMAT na internet⁵, opção "Dissertações", cujo foco está nas funções envolvidas, no ensino do próprio Método aos alunos do Ensino Médio ou mesmo na utilização do Método como apoio para o ensino de funções e/ou modelagem matemática.

Entretanto, diferentemente dos autores acima citados, e dos demais trabalhos disponíveis, este trabalho tem foco em uma Abordagem Computacional buscando não apenas apresentar o Método de Lagrange, mas também propõe uma forma de implementá-lo através de planilha eletrônica, algoritmo em linguagem natural (português não estruturado) e programa em linguagem de computador, além de apresentar duas abordagens diferentes (direta e indireta) e dois tipos de cenários a serem resolvidos (normal e inverso).

Em comum, todos os trabalhos citados afirmam que um Método de Interpolação Polinomial é uma forma de determinar uma função polinomial $p(x) = y$, tal que $p(x_0) = y_0$, $p(x_1) = y_1$, $p(x_2) = y_2$, $p(x_3) = y_3$, ..., $p(x_n) = y_n$, onde os pares ordenados (x_i, y_i) , $i = 0$ a n , são pontos **distintos** previamente conhecidos e que precisam todos, sem exceção, atender a regra de relação definida por $p(x)$.

Como $p(x)$ é uma função polinomial, a sua aparência é conhecida⁶:

$$p(x) = a_m x^m + a_{m-1} x^{m-1} + \dots + a_1 x + a_0. \quad (2.1)$$

Naturalmente, surge uma pergunta: e para todo e qualquer conjunto de pontos distintos, pode-se esperar encontrar um polinômio interpolador? A resposta é **sim**. MARTINS [5], apresenta uma demonstração da existência e da unicidade de um polinômio interpolador através de um sistema linear. Na seção 2.4 mais adiante será apresentada outra forma de demonstrar a unicidade deste polinômio interpolador.

Avalie-se o seguinte cenário: uma função f que descreve o volume de líquido em um tanque ao longo do tempo, $V = f(t)$, que está sendo esvaziado pela base apenas pela ação da gravidade. Percebe-se facilmente que a taxa de vazão diminui ao longo do tempo uma vez que a massa de líquido fazendo pressão sobre o ponto de vazamento diminui

⁵ Endereço eletrônico do Mestrado Profissional em Matemática em Rede Nacional – PROFMAT: <https://profmat-sbm.org.br/>

⁶ Mais adiante discutiremos qual o grau do polinômio gerado pela Interpolação Polinomial.

ao longo do tempo. Portanto, o gráfico desta função não será uma reta (posto que a taxa de vazão não é constante) e esta função será monótona e continuamente decrescente, comportamento este que pode ser apresentado por uma função polinomial, desde que observe-se um intervalo restrito.

Além do mais, considere-se que, por se tratar de uma função, é necessário que haja uma relação de dependência entre y e x de tal forma que **a variável y tem que depender do valor da variável x** , não sendo possível que situações erráticas sejam interpoladas. Por exemplo, se os pontos conhecidos correlacionarem o salário de uma pessoa com a sua idade, esses pontos podem ser representados dentro de um plano cartesiano com eixos *idade* e *salário*, mas estarão espalhados de forma errática e este não é um cenário onde uma interpolação seja aplicável, pois simplesmente não é representável por uma função, já que o salário não depende da idade da pessoa.

O ponto ao qual se quer chegar é que não basta existir um conjunto de pontos distintos e conhecidos para que seja possível utilizar uma interpolação (seja polinomial ou não). Portanto, antes de lançar mão da interpolação polinomial é necessário analisar em que contexto o cenário a ser estudado se encontra. No caso do tanque de líquido citado acima, o eixo- x é o tempo e o eixo- y é o volume e, portanto, não se identifica um comportamento errático, mas é necessário ter a certeza de que existe uma relação de dependência direta entre o volume e o tempo, sem existir qualquer outra variável interferindo no volume, para que se possa haver interesse no uso do método de interpolação.

No caso particular do exemplo do tanque, uma maneira de validar se o cenário é apropriado para o uso do método de interpolação é usar esses pontos para calcular o polinômio interpolador e, caso este polinômio encontrado não seja monótono e decrescente no intervalo dado (de volume Total no tempo zero até volume Zero no tempo final), então, tal cenário do mundo real não pode ser estudado através de interpolação polinomial, pois um comportamento de 'sobe e desce' não é aceitável, uma vez que no cenário colocado, em nenhum momento é possível haver um aumento de volume.

Pode-se usar o polinômio interpolador para calcular a ordenada de qualquer abscissa?

Se considerar que os pontos conhecidos delimitam um intervalo $[x_0, x_n]$ dentro do qual todos os demais pontos são encontrados, extrapolar a validade do polinômio interpolador para fora desse intervalo pode ser um equívoco e pode levar o estudante a erro. Num cenário do mundo real, por exemplo, ao avaliar o comportamento de uma máquina a partir de um momento inicial no tempo, até um momento final no tempo, espera-se que o polinômio encontrado represente o padrão de comportamento dessa máquina *nesse intervalo de tempo* e nada garante que a máquina continuará se comportando neste padrão após o término desta avaliação. Portanto, o ideal é que a função polinomial obtida através do método de interpolação polinomial somente seja usada para determinar o valor aproximado de y para um valor de x que pertença ao intervalo compreendido entre os pontos extremos,

que serão os de menor e maior abscissa, isto porque o cenário que está sendo avaliado ocorreu somente neste intervalo da variável independente e o comportamento da função não pode ser estendido para além dessas fronteiras. No caso do exemplo da figura 2, este intervalo é representado pelos pontos de abscissa 0 e abscissa 6. Não foi à toa que nas figuras 1 e 2 o gráfico da função está limitado a esse intervalo. Claro que pode haver casos em que as circunstâncias permitam que essa extrapolação seja feita sem risco de perda de confiança, mas esse cenário deve ser usado como exceção e não como regra.

O Método de Interpolação Polinomial de Lagrange aparece em sua obra "*Leçons élémentaires sur les mathématiques*" (Aulas de Matemática Elementar, em tradução livre) [8], onde o autor discorre sobre vários assuntos ao longo de 5 Capítulos, chamados de "*Lectures*" ou Leituras.

RIBNIKOV [9] em seu livro "A História da Matemática", afirma que a "influência direta dos problemas das ciências naturais exatas no desenvolvimento da Matemática" produziu ao longo da história uma quantidade enorme de teorias, conceitos e métodos. Um exemplo dessa influência que o autor russo cita é visto na obra de Lagrange, mais especificamente na Leitura V, cujo título é "O Emprego de Curvas na Solução de Problemas", e no qual afirma: "Esses dois exemplos podem ser suficientes para ilustrar a utilidade do método das curvas para resolver problemas".

Nesta Leitura V, Lagrange detalha melhor:

A questão é, de fato, redutível à descrição de uma curva que passará por um certo número de pontos, sejam esses pontos obtidos por cálculo ou construção ou mesmo por observação ou simples experiências totalmente independentes uma das outras. O problema é, na verdade, indeterminado, uma vez que passando por um certo número de pontos pode haver uma infinidade de diferentes curvas, regulares ou irregulares e que são sujeitos a equações ou mesmo traçadas manualmente. Mas a questão não é encontrar uma solução qualquer, mas a mais simples e mais fácil.

Portanto, se forem dados apenas dois pontos, a solução mais simples será uma linha reta ligando esses dois pontos. Se forem dados três pontos, um arco circular será a linha mais simples possível de ser descrita.

Deve-se observar que, de acordo com o Primeiro Axioma da Incidência [10], de Euclides, dois pontos distintos determinam uma única reta, o que não impede que se afirme que por dois pontos distintos podem ser traçadas infinitas curvas. Mas Lagrange usa os adjetivos **simples** e **fácil** e, por isso, uma reta é a curva⁷ utilizada para dois pontos distintos⁸.

Lagrange continua:

Mas se o círculo é a curva mais simples de se descrever, não o é com relação à equação entre suas abscissas e ordenadas. Deste ponto de vista as curvas que podem ser consideradas como as mais simples são aquelas cujas ordenadas podem ser expressas por uma função racional a partir de suas abscissas, e

⁷ Não podemos ignorar o fato de que uma reta é um caso particular de curva, onde a inclinação é constante.

⁸ Por que dois pontos distintos? Porque, em contraposição, podemos ter dois pontos coincidentes (de mesmas coordenadas) que, na prática, acabam por determinar um único ponto no plano (ou no espaço) pelo qual podemos traçar infinitas retas. Esse rigor matemático é necessário.

representada pela seguinte equação $y = a + bx + cx^2 + dx^3 + \dots$, onde y é a ordenada e x a abscissa. Curvas desse tipo são geralmente chamadas de parabólicas porque podem ser consideradas como uma generalização da parábola – uma curva representada pela equação acima exposta quando possui apenas os três primeiros termos. Nós já mostramos o emprego na resolução de equações [...] por essa razão que uma curva desse tipo pode sempre ser usada para passar por tantos pontos quanto quisermos – sendo necessário, apenas calcular tantos coeficientes a, b, c, \dots quantos forem os pontos dados.

Aqui Lagrange dá a dica de que, se houver n pontos dados, haverá n coeficientes, inclusive aquele do termo independente ou de grau 0. Isso implica que o grau do polinômio calculado será $n - 1$. Observemos que tal afirmação é equivalente a dizer que, para $n + 1$ pontos, o polinômio será de grau n .

Na continuação, Lagrange afirma que “Newton foi o primeiro a propor este problema. O que se segue é a solução dada por ele”, e faz a demonstração da solução dada por Newton e que é conhecida como Método de Interpolação Polinomial de Newton [11]⁹.

Na sequência, Lagrange faz a seguinte afirmação sobre a solução dada por Newton: “Mas esta solução pode ser simplificada pelas seguintes considerações” e começa a expor seu método.

2.3 O Desenvolvimento do Método

O raciocínio de Lagrange foi o seguinte:

Sejam $x_i, i = 0$ a $n, (n + 1)$ abscissas distintas.

Sejam $y_i, i = 0$ a n , as respectivas imagens de x_i , sendo $y_i = f(x_i)$.

Portanto, há $(n + 1)$ pontos previamente definidos, da forma $(x_i, f(x_i))$.

Seja $p_n(x)$ o polinômio de grau $\leq n$ que interpola os $(n + 1)$ pontos dados.

Então, $p_n(x)$ pode ser escrito como uma Combinação Linear de $(n + 1)$ polinômios de grau n , o que gera:

$$p_n(x) = y_0 L_0(x) + y_1 L_1(x) + \dots + y_n L_n(x), \text{ onde } L_i(x) \text{ são polinômios de grau } n.$$

Mas, se por hipótese, cada um dos $(n + 1)$ pontos resolvem $p_n(x)$, então pode-se afirmar que $p_n(x_i) = y_i$, o que significa que

$$p_n(x_i) = y_0 L_0(x_i) + y_1 L_1(x_i) + \dots + y_n L_n(x_i) = y_i. \quad (2.2)$$

Desenvolvendo a Equação (3.1) acima para cada um dos i , de 0 a n , são encontrados:

$$\begin{aligned} \text{para } i = 0, p_n(x_0) &= y_0 L_0(x_0) + y_1 L_1(x_0) + \dots + y_n L_n(x_0) = y_0 \\ \text{logo } L_0(x_0) &= 1 \text{ e os demais } L(x) \text{ serão iguais a } 0; \end{aligned} \quad (2.3)$$

⁹ **Sugestão de Pesquisa:** Newton usou um método chamado de Diferenças Divididas.

$$\begin{aligned} \text{para } i = 1, p_n(x_1) &= y_0 L_0(x_1) + y_1 L_1(x_1) + \dots + y_n L_n(x_1) = y_1 \\ \text{logo } L_1(x_1) &= 1 \text{ e os demais } L(x) \text{ serão iguais a } 0; \end{aligned} \quad (2.4)$$

$$\begin{aligned} \text{para } i = n, p_n(x_n) &= y_0 L_0(x_n) + y_1 L_1(x_n) + \dots + y_n L_n(x_n) = y_n \\ \text{logo } L_n(x_n) &= 1 \text{ e os demais } L(x) \text{ serão iguais a } 0. \end{aligned} \quad (2.5)$$

Resumindo:

$$L_k(x_i) = \begin{cases} 0 & \text{se } k \neq i \\ 1 & \text{se } k = i. \end{cases} \quad (2.6)$$

Agora, se $L(x)$ é um polinômio de grau n^{10} , então ele terá n raízes e poderá ser escrito na forma fatorada. Ou seja, se $r_i, i = 1$ a n , forem considerados como sendo as raízes de $L(x)$, então:

$$L(x) = (x - r_1)(x - r_2)(x - r_3) \dots (x - r_n). \quad (2.7)$$

Mas já é sabido que $L_0(x) = 0$ para $x = x_1, x_2, \dots, x_n$; portanto, x_1, x_2, \dots, x_n são raízes de $L_0(x)$. Logo,

$$L_0(x) = (x - x_1)(x - x_2)(x - x_3) \dots (x - x_n). \quad (2.8)$$

Mas é sabido também que $L_0(x) = 1$ quando $x = x_0$. Então, se for colocado um denominador que seja igual ao numerador quando $x = x_0$, o denominador ‘cancelará’ todo o numerador, resultando em 1.

Portanto,

$$L_0(x) = \frac{(x - x_1)(x - x_2) \dots (x - x_n)}{(x_0 - x_1)(x_0 - x_2) \dots (x_0 - x_n)}. \quad (2.9)$$

¹⁰ $L(x)$ foi convenientemente escolhido como um polinômio de grau n e coeficiente 1 no termo de grau n .

Agora, é possível verificar se ambas as regras da equação 2.6 foram satisfeitas:

- se x assumir qualquer valor de x_1 a x_n , um dos fatores do numerador assumirá o valor zero, levando a zero o numerador e, por conseguinte, fazendo $L_0(x)$ ficar zerado, como esperado; como, neste caso, nenhum fator do denominador vai a zero, não há qualquer risco de uma indeterminação ser encontrada¹¹;
- se x assumir o valor de x_0 , então o numerador ficará igual ao denominador, levando $L_0(x)$ a assumir o valor 1.

Algum leitor, nesse momento, pode se questionar para que serve, então, essa expressão que assumirá 0 ou 1 dependendo do valor de x . A resposta é simples: o valor 0 ou 1 será atingido nos pontos já conhecidos (abscissas x_0 a x_n) e levará todo o polinômio $p_n(x)$ ao valor da ordenada y correspondente. Porém para abscissas de pontos que estejam fora do conjunto de pontos conhecidos os valores de $L_i(x)$ não serão 0 ou 1 (até podem, eventualmente, sê-lo), levando o polinômio $p_n(x)$ ao valor da respectiva ordenada que se quer calcular, processo esse, aliás, que representa, justamente, o objetivo de encontrarmos o polinômio $p_n(x)$ e do próprio método de interpolação polinomial: **achar a ordenada de pontos não conhecidos a partir de suas abscissas**.

Uma vez calculado $L_0(x)$, raciocínio análogo pode ser usado para calcular os demais $L_k(x)$.

Dessa forma encontram-se:

$$L_0(x) = \frac{(x - x_1)(x - x_2) \dots (x - x_n)}{(x_0 - x_1)(x_0 - x_2) \dots (x_0 - x_n)} \quad (2.10)$$

$$L_1(x) = \frac{(x - x_0)(x - x_2) \dots (x - x_n)}{(x_1 - x_0)(x_1 - x_2) \dots (x_1 - x_n)} \quad (2.11)$$

$$L_n(x) = \frac{(x - x_0)(x - x_1) \dots (x - x_{n-1})}{(x_n - x_0)(x_n - x_1) \dots (x_n - x_{n-1})} \quad (2.12)$$

¹¹ Devemos observar que, para que o denominador possa assumir um valor zerado, é necessário que x_0 seja igual a um dos outros x_i , o que não é possível uma vez que as raízes precisam ser todas diferentes entre si. Essa afirmação é válida dentro do contexto do problema, uma vez que, se tivermos duas raízes iguais, então temos, na verdade, dois pontos iguais no conjunto de pontos conhecidos, o que não faria sentido, repito, dentro deste contexto. Nesse caso, teríamos um ponto a menos dentro do conjunto e o raciocínio permanece totalmente válido e coerente considerando este novo conjunto de pontos.

Cada uma dessas expressões terá n fatores no numerador e no denominador, onde n é a quantidade de pontos conhecidos, menos 1, ou seja, $((n + 1) - 1 = n)$.

A partir da Equação 2.2 e aproveitando os resultados encontrados acima, surge a seguinte expressão final para $P(x)$, o polinômio interpolador:

Fórmula Geral (FG) do Polinômio Interpolador

$$p(x) = y_0 \frac{(x - x_1)(x - x_2) \dots (x - x_n)}{(x_0 - x_1)(x_0 - x_2) \dots (x_0 - x_n)} + y_1 \frac{(x - x_0)(x - x_2) \dots (x - x_n)}{(x_1 - x_0)(x_1 - x_2) \dots (x_1 - x_n)} + \dots + y_n \frac{(x - x_0)(x - x_1) \dots (x - x_{n-1})}{(x_n - x_0)(x_n - x_1) \dots (x_n - x_{n-1})}. \quad (2.13)$$

2.4 Estudo da Unicidade do Polinômio Interpolador

Uma vez encontrado o polinômio interpolador, surge a necessidade de verificar se este polinômio encontrado é o **único** polinômio interpolador ou se é **somente um dentre vários** polinômios interpoladores existentes. É evidente que conjuntos diferentes de pontos conhecidos podem levar a um mesmo polinômio, uma vez que há infinitos pontos pelos quais o polinômio passa; mas a questão é se um mesmo conjunto de pontos conhecidos pode levar a mais de um polinômio.

O que é preciso provar é:

“Se há um conjunto de $n + 1$ pontos (x, y) distintos conhecidos então existe um, e somente um, polinômio interpolador de grau $\leq n$ que passa por todos os $n + 1$ pontos distintos conhecidos”.

Usando o método da *redução ao absurdo*, parte-se da **negação** da tese:

“Existe mais de um polinômio interpolador de grau $\leq n$ que passa por todos os $n + 1$ pontos conhecidos”.

Neste caso, sejam $p_n(x)$ e $q_n(x)$ dois polinômios de grau $\leq n$ e que, por serem ambos interpoladores a partir de um conjunto de $n + 1$ pontos distintos conhecidos, assumirão **os mesmos valores** para cada um desses $n + 1$ pontos distintos.

Seja o polinômio $s(x) = p_n(x) - q_n(x)$. Como ambos os polinômios p e q são de grau $\leq n$, o polinômio s também será de grau $\leq n$. Entretanto, se os $n + 1$ pontos distintos geram valores idênticos para p e para q , então $s_n(x) = 0$ para todos esses $n + 1$ pontos distintos. Portanto, **esses $n + 1$ pontos distintos são raízes de s** , o que é um absurdo pois, segundo ensina LIMA [12], *“uma função polinomial de grau n não pode ter mais do que n raízes”*. Então, $p_n(x) = q_n(x)$ e $s_n(x)$ é a função polinomial chamada de *identicamente nula*, pois $s_n(x) = 0$ para todos os pontos do domínio.

Assim, fica provada a unicidade do polinômio interpolador.

3 DESENVOLVIMENTO ALGÉBRICO DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE

3.1 O Polinômio Interpolador para um Único Ponto Conhecido

É sabido que por um único ponto é possível passar infinitas retas.

E será que o método de interpolação polinomial de Lagrange é coerente com essa regra? Como este método se comportará se for considerado apenas um único ponto?

Se houver, então, apenas x_0 e $y_0 = f(x_0)$.

Aplicando o método de Lagrange, a partir da Fórmula Geral (Equação 2.13), e considerando que só temos 1 ponto, é possível afirmar que:

$$p_0(x) = y_0 L_0(x) \quad (3.1)$$

Ao tentar calcular $L_0(x)$, a partir da Equação 2.9, percebe-se que não existem as abscissas x_1, x_2 , etc e, portanto, o numerador ficará igual a zero. Pelo mesmo raciocínio, chega-se a um denominador também igual a zero.

Então:

$$p_0(x) = y_0 \frac{0}{0} = \text{INDETERMINAÇÃO} \quad (3.2)$$

Conclusão: se não há como determinar $p_n(x)$, então qualquer polinômio serve, inclusive os de grau 1 (retas), desde que $p(x_0) = y_0$.

Evidentemente, é possível expandir essa conclusão para qualquer função $f(x)$, mesmo que não seja polinomial, desde que $f(x_0) = y_0$

3.2 O Polinômio Interpolador para Dois Pontos Distintos Conhecidos

Pode-se aplicar novamente o método visto acima, agora para determinar o polinômio interpolador quando há 2 pontos distintos conhecidos.

Neste caso, existem x_0 e x_1 , sendo $y_0 = f(x_0)$ e $y_1 = f(x_1)$.

Aplicando o método de Lagrange, a partir da Fórmula Geral (Equação 2.13), e considerando que há 2 pontos distintos:

$$p_1(x) = y_0 L_0(x) + y_1 L_1(x). \quad (3.3)$$

Pela Fórmula Geral:

$$L_0(x) = \frac{(x - x_1)}{(x_0 - x_1)}. \quad (3.4)$$

$$L_1(x) = \frac{(x - x_0)}{(x_1 - x_0)}. \quad (3.5)$$

O que gera:

$$p_1(x) = y_0 \frac{(x - x_1)}{(x_0 - x_1)} + y_1 \frac{(x - x_0)}{(x_1 - x_0)} = \frac{y_0(x - x_1) - y_1(x - x_0)}{(x_0 - x_1)}. \quad (3.6)$$

$$p_1(x) = \frac{y_0x - y_0x_1 - y_1x + y_1x_0}{(x_0 - x_1)}. \quad (3.7)$$

Agrupando os termos em x:

$$p_1(x) = \frac{x(y_0 - y_1) - y_0x_1 + y_1x_0}{x_0 - x_1}. \quad (3.8)$$

Fazendo algumas substituições nas expressões que contêm os valores conhecidos x_0 e x_1 , y_0 e y_1 , da seguinte forma:

$$a = \frac{y_0 - y_1}{x_0 - x_1} = \frac{\Delta y}{\Delta x}. \quad (3.9)$$

$$b = \frac{y_1x_0 - y_0x_1}{x_0 - x_1}. \quad (3.10)$$

Retornando à expressão do polinômio será encontrada a expressão:

$$p_1(x) = ax + b. \quad (3.11)$$

Que é a expressão geral de uma reta, que possui grau 1, com **a** indicando sua inclinação (coeficiente angular) e **b** o coeficiente linear (ponto em que a reta intercepta o eixo-y)¹².

¹² Sugestão: substituir x por x₀ e trabalhar algebricamente a expressão de p_n; tem que chegar em y₀. Isso também vale para x₁, tendo que chegar em y₁

3.3 O Polinômio Interpolador para Três Pontos Distintos Conhecidos

Desta vez, será aplicado o método visto acima para determinar o polinômio interpolador quando há 3 pontos distintos conhecidos.

Neste caso, existem x_0 , x_1 e x_2 sendo $y_0 = f(x_0)$, $y_1 = f(x_1)$ e $y_2 = f(x_2)$.

Aplicando o método de Lagrange, a partir da Fórmula Geral (Equação 2.13), e considerando que desta vez há 3 pontos:

$$p_2(x) = y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x). \quad (3.12)$$

Pela Fórmula Geral:

$$L_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}. \quad (3.13)$$

$$L_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}. \quad (3.14)$$

$$L_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}. \quad (3.15)$$

O que gera:

$$p_2(x) = y_0 \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + y_1 \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + y_2 \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}. \quad (3.16)$$

Para facilitar o desenvolvimento algébrico dessa expressão, podem ser aplicadas desde já algumas substituições, considerando que os denominadores dos três termos são valores constantes, pois não trazem a variável x em sua expressão, mas apenas as coordenadas dos três pontos conhecidos: x_0 , x_1 , x_2 , y_0 , y_1 , y_2 :

$$q = (x_0 - x_1)(x_0 - x_2). \quad (3.17)$$

$$r = (x_1 - x_0)(x_1 - x_2). \quad (3.18)$$

$$s = (x_2 - x_0)(x_2 - x_1). \quad (3.19)$$

$$\begin{aligned}
 p_2(x) &= y_0 \frac{(x-x_1)(x-x_2)}{q} + y_1 \frac{(x-x_0)(x-x_2)}{r} + y_2 \frac{(x-x_0)(x-x_1)}{s} = \\
 &= \frac{(y_0rs(x^2 - xx_1 - xx_2 + x_1x_2) + y_1qs(x^2 - xx_0 - xx_2 + x_0x_2) + y_2qr(x^2 - xx_0 - xx_1 + x_0x_1))}{qrs}.
 \end{aligned}
 \tag{3.20}$$

Agrupando os termos em x e separando as frações:

$$\begin{aligned}
 p_2(x) &= \frac{x^2(y_0rs + y_1qs + y_2qr)}{qrs} \\
 &\quad - \frac{x[y_0rs(x_1 + x_2) + y_1qs(x_0 + x_2) + y_2qr(x_0 + x_1)]}{qrs} \\
 &\quad + \frac{y_0rsx_1x_2 + y_1qsx_0x_2 + y_2qrx_0x_1}{qrs}.
 \end{aligned}
 \tag{3.21}$$

Fazendo algumas outras substituições nas expressões que contêm os valores conhecidos x_0 , x_1 , x_2 , y_0 , y_1 , y_2 :

$$a = \frac{y_0rs + y_1qs + y_2qr}{qrs}.
 \tag{3.22}$$

$$b = \frac{y_0rs(x_1 + x_2) + y_1qs(x_0 + x_2) + y_2qr(x_0 + x_1)}{qrs}.
 \tag{3.23}$$

$$c = \frac{y_0rsx_1x_2 + y_1qsx_0x_2 + y_2qrx_0x_1}{qrs}.
 \tag{3.24}$$

Retornando à expressão do polinômio será encontrada a expressão:

$$p_2(x) = ax^2 + bx + c.
 \tag{3.25}$$

Que é a expressão geral de uma parábola, que possui grau 2, com a indicando sua concavidade e c o coeficiente linear (ponto em que a parábola intercepta o eixo-y)¹³.

¹³ Sugestão: substituir x por x₀ e trabalhar algebricamente a expressão de p_n; tem que chegar em y₀. O que também vale para x₁ e x₂, tendo que chegar em y₁ e y₂, respectivamente.

3.4 Estudo do Grau do Polinômio Interpolador

É possível perceber que todas as referências feitas ao polinômio interpolador até o momento afirmam que seu grau é MENOR OU IGUAL a n , e não IGUAL a n .

Nas duas seções anteriores foi feito um desenvolvimento algébrico partindo da Fórmula Geral do método aqui estudado, e foi possível constatar que quando há 2 pontos distintos será encontrada uma reta (que possui grau 1) e quando há 3 pontos distintos será encontrada uma parábola (que possui grau 2).

Além disso, a Fórmula Geral (Equação 2.13) apresentada ao final da seção 2.3, deixa claro que para $n + 1$ pontos distintos conhecidos, há n termos no polinômio p_x e que cada um desses termos possui n fatores em seus denominadores e que cada um dos fatores apresenta a variável x . Portanto, a julgar por todas essas evidências, seria natural chegar à conclusão de que o polinômio interpolador possui **grau “= n ” e não “ $\leq n$ ”**.

Então, por que o grau do polinômio interpolador é “ $\leq n$ ”?

Para responder essa pergunta é necessário lembrar da afirmação de Lagrange no início da apresentação do seu método: “Mas a questão não é encontrar uma solução qualquer, mas a mais simples e mais fácil”.

Lembre-se que se houver 2 pontos distintos é possível traçar infinitas **curvas** (não retas) que passam por esses dois pontos¹⁴ (soluções quaisquer), mas a curva calculada será uma reta (solução mais simples).

Uma vez definida essa reta, que é a curva mais simples e que passa por esses dois pontos A e B (veja a figura 3), pode-se escolher sobre ela um terceiro ponto C. Agora há 3 pontos distintos mas, por serem **colineares**, a curva mais simples que passa por eles é uma reta¹⁵. E se for aplicado o Método de Lagrange ele produzirá essa mesma reta. Portanto, nesses casos, apesar de 3 pontos distintos ($n + 1 = 3$ $n = 2$) o polinômio interpolador será de grau 1 (menor que n) e não grau 2 (igual a n).

¹⁴ Se não ficou clara essa afirmação, imaginemos uma parábola e escolha dois pontos quaisquer e distintos sobre o seu gráfico e trace uma reta ligando esses dois pontos ou imagine uma parábola de concavidade invertida, ou mesmo uma função polinomial cuja parábola inicialmente descrita seja apenas um pedaço, ou mesmo um arco circular passando por esses dois pontos. Tudo isso são apenas exemplos para mostrar que há, sim, infinitas

¹⁵ Até porque não poderia jamais ser uma parábola posto que são colineares. . .

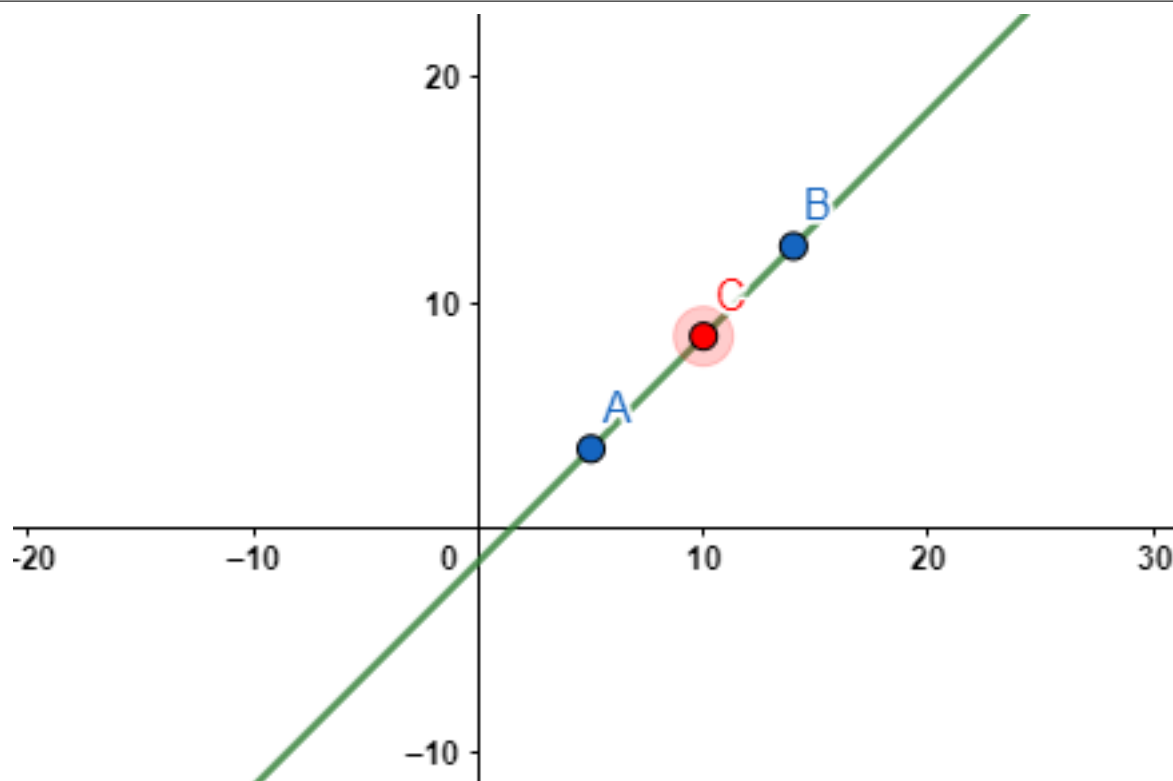


Figura 3 – Exemplo de 3 pontos distintos que, por serem colineares, serão interpolados através de uma reta e não de uma parábola.

Raciocínio análogo pode ser utilizado em qualquer quantidade de pontos sobre uma mesma reta (por exemplo, se for escolhido um quarto ponto sobre essa reta, o que fará com que n seja 3, continuar sendo gerada uma reta) ou uma quantidade qualquer $n + 1$ de pontos distintos que podem estar todos sobre uma mesma curva de grau menor que n fazendo com que o polinômio interpolador calculado tenha o grau dessa curva e não o grau n que a Fórmula Geral dá a entender que tenha.

No Capítulo seguinte serão resolvidos alguns exemplos numéricos e tal raciocínio será validado.

4 DESENVOLVIMENTO NUMÉRICO DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE

Como foi dito anteriormente, o objetivo final de um Método de Interpolação é permitir que, a partir de um conjunto limitado de pontos distintos conhecidos, seja possível determinar qualquer ponto fora desse conjunto e que guarde relação com os já conhecidos segundo um comportamento definido a partir deste conjunto de pontos, comportamento este ditado por uma função interpoladora.

No caso do Método de Lagrange essa função interpoladora é um polinômio.

Neste capítulo serão descritas duas abordagens diferentes de uso do Método de Lagrange, que aqui são chamadas de **Forma Direta** e **Forma Indireta**.

4.1 Forma Direta para Cálculo da Imagem de um Ponto Não Conhecido

A **Forma Direta** se dá quando é usada a Fórmula Geral (Equação 2.13) que foi apresentada ao final da seção 2.3 na qual são aplicadas diretamente as coordenadas dos pontos já conhecidos e colocado o valor da abscissa do ponto desconhecido diretamente na variável x e, assim, o valor de sua ordenada é calculada.

Seguem dois exemplos numéricos para que a aplicação da Forma Direta possa ser entendida:

a) **Dois** pontos A (5 , 4) e B (14 , 13) conhecidos, deseja-se calcular a ordenada do ponto de abscissa 10^{16} .

Considerando o ponto A como sendo o ponto 0 e o ponto B como sendo o ponto 1^{17} são feitas as seguintes substituições:

$$x_0 = 5 \text{ e } y_0 = 4.$$

$$x_1 = 14 \text{ e } y_1 = 13.$$

$$x = 10.$$

$$p(10) = y = 4 \frac{(10-14)}{(5-14)} + 13 \frac{(10-5)}{(14-5)} = 4 \frac{-4}{-9} + 13 \frac{5}{9} = \frac{16}{9} + \frac{65}{9} = \frac{81}{9} = 9.$$

Portanto, a ordenada do ponto de abscissa 10 será 9 quando os dois pontos dados são interpolados. De fato, o ponto (10 , 9) pertence à reta $y = x - 1$. São os pontos usados na figura 3.

b) **Três** pontos A (0,5 , 1,25) , B(1,5 , -0,75) e C(4 , 3) conhecidos, deseja-se calcular a ordenada do ponto de abscissa 3^{18} .

Serão considerados o ponto A como o ponto 0 , o ponto B como o ponto 1 e o ponto C como o ponto 2. Observe-se que a escolha de qual ponto será o 0, o 1 e o 2 é irrelevante

¹⁶ Para esse exemplo, foi usada a função $f(x) = x - 1$ à qual ambos os pontos pertencem. Saber a função vai facilitar a validação do cálculo realizado.

¹⁷ Sugestão de Estudo Adicional: verificar que não importa qual ponto será considerado como 0 ou como 1.

¹⁸ Para esse exemplo, foi usada a função $f(x) = x^2 - 4x + 3$.

para a obtenção do resultado esperado.

Assim, realizando as substituições adequadas, encontra-se:

$$x_0 = 0,5 \text{ e } y_0 = 1,25.$$

$$x_1 = 1,5 \text{ e } y_1 = -0,75.$$

$$x_2 = 4 \text{ e } y_2 = 3.$$

$$x = 3.$$

$$p(0) = 1,25 \frac{(3-1,5)(3-4)}{(0,5-1,5)(0,5-4)} + (-0,75) \frac{(3-0,5)(3-4)}{(1,5-0,5)(1,5-4)} + 3 \frac{(3-0,5)(3-1,5)}{(4-0,5)(4-1,5)}.$$

$$p(0) = 1,25 \frac{(1,5)(-1)}{(-1)(-3,5)} + (-0,75) \frac{(2,5)(-1)}{(1)(-2,5)} + 3 \frac{(2,5)(1,5)}{(3,5)(2,5)}.$$

$$p(0) = \frac{(-1,875)}{3,5} - 0,75 + 3 \frac{1,5}{3,5} = \frac{-1,875-2,625+4,5}{3,5} = 0.$$

Portanto, a ordenada do ponto de abscissa 3 será 0 quando os três pontos dados são interpolados. De fato, o ponto (3, 0) pertence à parábola $y = x^2 - 4x + 3$ (figura abaixo).

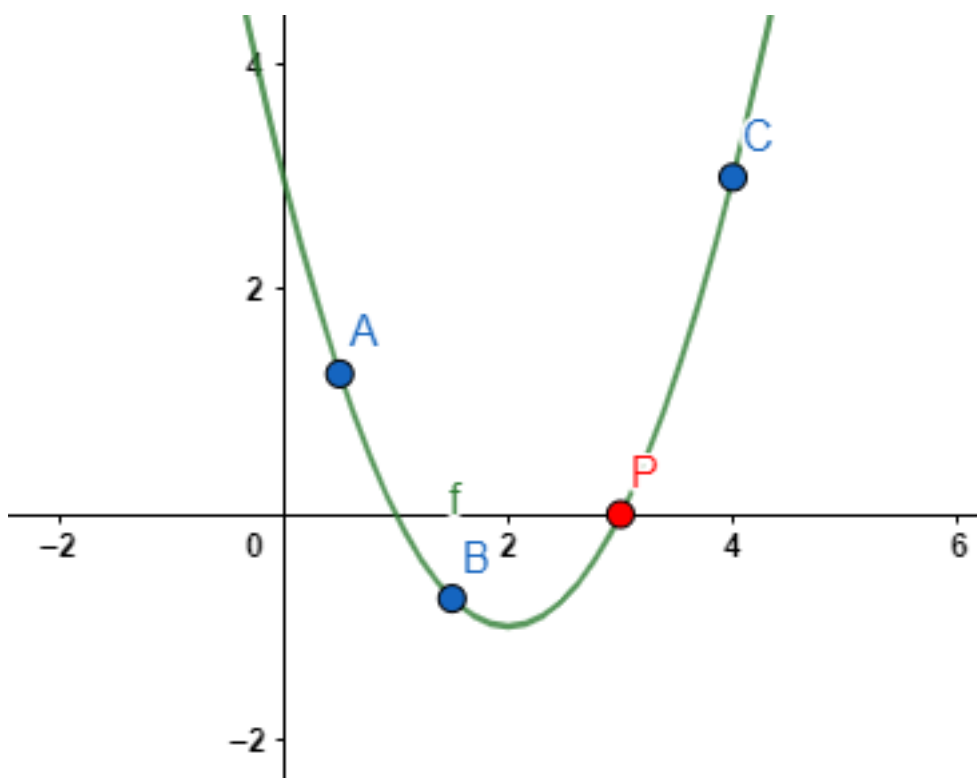


Figura 4 – Parábola com os três pontos distintos A, B e C conhecidos (em azul) e o ponto P a descobrir (em vermelho)

4.2 Forma Indireta para Cálculo da Imagem de um Ponto Não Conhecido

A **Forma Indireta** ocorre quando é usada a Fórmula Geral (Equação 2.13) apresentada ao final da seção 2.3 e nela são diretamente aplicadas as coordenadas dos

pontos já conhecidos e trabalhada algebricamente a expressão resultante até que seja encontrado o Polinômio Interpolador $p_n(x)$ e, sobre este, a variável x é substituída pelo valor da abscissa do ponto desconhecido para, então, poder ser calculado o valor de sua ordenada.

A vantagem da forma indireta é que, quando os pontos conhecidos são estáticos, isto é, não mudam (pelo menos não com bastante frequência), a aplicação da Fórmula Geral, que é tão mais trabalhosa algebricamente quanto maior for a quantidade de números conhecidos, só é feita uma única vez, resultando em $p_n(x)$ que, por sua vez, é muito mais facilmente utilizável para calcular a imagem do(s) ponto(s) desconhecidos. Além disso, como será visto mais adiante, ainda neste trabalho, para o problema inverso, a forma indireta terá que ser usada pois será necessário conhecer a expressão do polinômio interpolador.

Por outro lado, a forma indireta não se mostra mais vantajosa frente à direta nos cenários em que os pontos conhecidos variam continuamente pois, nestes casos, ambas as formas são semelhantes do ponto de vista do esforço, sendo que a forma direta se mostra um pouco mais ‘amigável’ pois já se aplica em x o valor da abscissa do ponto desconhecido e já se eliminam todas as variáveis do processo.

Aplicando os mesmos exemplos numéricos da seção anterior, agora na forma indireta:

a) **Dois** pontos A (5 , 4) e B (14 , 13) conhecidos, deseja-se calcular a ordenada do ponto de abscissa 10.

Chamando A de ponto 0 e B de ponto 1:

$$x_0 = 5 \text{ e } y_0 = 4.$$

$$x_1 = 14 \text{ e } y_1 = 13.$$

$$p(x) = 4 \frac{(x-14)}{(5-14)} + 13 \frac{(x-5)}{(14-5)} = 4 \frac{(x-14)}{(-9)} + 13 \frac{(x-5)}{9}.$$

$$p(x) = \frac{(-4)(x-14)}{9} + 13 \frac{(x-5)}{9} = \frac{-4x+56+13x-65}{9} = \frac{9x-9}{9}.$$

$$p(x) = x - 1.$$

Chega-se à mesma expressão para $p(x)$ já conhecida como sendo a correta, lá da seção 4.1. Substituindo x por 10 na expressão $p(x)$, é encontrado $y = 9$, chegando-se ao mesmo ponto (10 , 9) encontrado na seção anterior. Importante destacar que, na forma indireta de cálculo, agora que $p(x)$ já é conhecido, pode-se aplicar a x qualquer valor que se queira sem que seja necessário refazer cálculos.

É interessante comentar que, no caso em que há dois pontos conhecidos, existe o método tradicional de se encontrar a expressão da reta que une esses dois pontos:

$$a = \frac{\Delta y}{\Delta x} = \frac{y_1 - y_0}{x_1 - x_0} = \frac{13 - 4}{14 - 5} = \frac{9}{9} = 1.$$

$$y = ax + b \implies y = x + b \implies b = y - x \implies b = 13 - 14 \implies b = -1.$$

$$y = x - 1.$$

Evidentemente, chega-se à mesma expressão da reta encontrada pelo Método de Lagrange. E não poderia ser diferente.

b) **Três** pontos A (0,5 , 1,25) , B (1,5 , -0,75) e C (4 , 3) conhecidos, deseja-se calcular a ordenada do ponto de abscissa 3.

Chamando A de ponto 0 , B de ponto 1 e C de ponto 2:

$$x_0 = 0,5 \text{ e } y_0 = 1,25.$$

$$x_1 = 1,5 \text{ e } y_1 = -0,75.$$

$$x_2 = 4 \text{ e } y_2 = 3.$$

$$p(x) = 1,25 \frac{(x-1,5)(x-4)}{(0,5-1,5)(0,5-4)} + (-0,75) \frac{(x-0,5)(x-4)}{(1,5-0,5)(1,5-4)} + 3 \frac{(x-0,5)(x-1,5)}{(4-0,5)(4-1,5)}.$$

$$p(x) = 1,25 \frac{(x-1,5)(x-4)}{(-1)(-3,5)} + (-0,75) \frac{(x-0,5)(x-4)}{(1)(-2,5)} + 3 \frac{(x-0,5)(x-1,5)}{(3,5)(2,5)}.$$

$$p(x) = \frac{1,25(x^2-5,5x+6)}{3,5} - \frac{0,75(x^2-4,5x+2)}{-2,5} + \frac{3(x^2-2x+0,75)}{8,75}.$$

$$p(x) = \frac{2,5(x^2-5,5x+6)}{7} + \frac{1,5(x^2-4,5x+2)}{5} + \frac{12(x^2-2x+0,75)}{35}.$$

$$p(x) = \frac{12,5(x^2-5,5x+6)}{35} + \frac{10,5(x^2-4,5x+2)}{35} + \frac{12(x^2-2x+0,75)}{35}.$$

$$p(x) = \frac{12,5x^2-68,75x+75+10,5x^2-47,25x+21+12x^2-24x+9}{35}.$$

$$p(x) = \frac{35x^2-140x+105}{35} \implies p(x) = x^2 - 4x + 3.$$

Mais uma vez, como esperado, chega-se à mesma expressão para $p(x)$ já conhecida como sendo a correta, lá da seção 4.1. Como aconteceu no exemplo anterior, agora que $p(x)$ já é conhecido, pode-se aplicar a x qualquer valor que se queira sem que seja necessário refazer cálculos. Substituindo x por 3 na expressão $p(x)$, é encontrado $y = 0$. Mesmo ponto (3, 0) encontrado na seção anterior.

4.3 Retorno ao Estudo do Grau do Polinômio Interpolador

Agora que já foram estudados dois exemplos numéricos é possível revalidar o que foi dito na seção 3.4 sobre o grau do polinômio interpolador poder ser menor que n e não precisar ser, obrigatoriamente, igual a n .

Para tanto, será usada a reta do exemplo (a) da seção anterior.

Como resultado do problema proposto, já são conhecidos três pontos pertencentes à reta dada: os dois dados pelo enunciado (A e B) e o terceiro calculado (C).

Usando a forma indireta para calcular $p_n(x)$ e tomando esses **três** pontos já conhecidos, o polinômio a ser encontrado tem que ser a mesma reta à qual os três pontos pertencem e cuja expressão já é conhecida.

São três os pontos conhecidos: A (5 , 4) e B (14 , 13) e C (10 , 9). O que se deseja é achar $p_n(x)$.

Chamando A de ponto 0 , B de ponto 1 e C de ponto 2, o desenvolvimento será da seguinte forma:

$$x_0 = 5 \text{ e } y_0 = 4.$$

$$x_1 = 14 \text{ e } y_1 = 13.$$

$$x_2 = 10 \text{ e } y_2 = 9.$$

$$p(x) = 4 \frac{(x-14)(x-10)}{(5-14)(5-10)} + 13 \frac{(x-5)(x-10)}{(14-5)(14-10)} + 9 \frac{(x-5)(x-14)}{(10-5)(10-14)}.$$

$$p(x) = 4 \frac{(x^2-24x+140)}{45} + 13 \frac{(x^2-15x+50)}{36} + 9 \frac{(x^2-19x+70)}{-20}.$$

$$p(x) = \frac{16(x^2-24x+140)+65(x^2-15x+50)-81(x^2-19x+70)}{180}.$$

$$p(x) = \frac{16x^2-384x+2240+65x^2-975x+3250-81x^2+1539x-5670}{180}.$$

$$p(x) = \frac{0x^2+180x-180}{180} \implies p(x) = x - 1.$$

Como esperado, apesar de haver 3 pontos distintos conhecidos, o polinômio interpolador tem grau 1 e não grau 2¹⁹.

¹⁹ Sugestão de Estudo: faça para a letra (b) a mesma verificação que foi feita para a letra (a) da seção 4.

5 DESENVOLVIMENTO COMPUTACIONAL DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE - FORMA DIRETA

O uso do computador nos dias de hoje está cada vez mais necessário, em praticamente todas as áreas profissionais e na vida pessoal. É muito difícil encontrar alguém que não tenha um celular no qual diversos aplicativos são executados, desde jogos até acesso às contas bancárias. Também é difícil encontrar alguém que não tenha um computador em casa.

Entretanto, nem todo mundo sabe desenvolver programas de computador. Isso requer um conhecimento técnico específico e muitos anos de estudo e prática.

Neste capítulo serão vistas algumas formas de desenvolver o Método de Lagrange no computador e tornar o método ainda mais útil e prático.

5.1 Cálculo pela Forma Direta Utilizando uma Planilha Eletrônica

Para aqueles que não sabem construir (desenvolver) programas de computador, há a opção de usar uma planilha eletrônica (existem diversas opções no mercado, inclusive gratuitas) para implementar o Método de Lagrange. O mais fácil é usar a forma direta para implementar este cálculo.

Neste trabalho será assumido que o leitor possui conhecimento suficiente de como uma planilha eletrônica funciona. Assim, não será necessário entrar em detalhes relacionados à ferramenta e será possível focar na implementação do método em si.

Para começar, será utilizado o primeiro cenário criado na seção 4.1.

a) Há **DOIS** pontos distintos conhecidos, e será adotada a forma direta:

Inicialmente, é necessário organizar a forma como os dados conhecidos serão dispostos na planilha. Os valores de y serão colocados na vertical (como um vetor coluna) e os valores de x na horizontal e na vertical (numa forma matricial $n \times n$). A matriz é necessária pois serão calculadas as diferenças entre as abscissas, duas a duas, por causa dos denominadores de $L(x)$. Usando os valores do item (a) da seção 4.1 produz-se:

8						
9		Valores de Y	Valores de X			
10	0	4,00		X0	1	
11	1	13,00		5,00	14,00	
12			0	5,00		
13			1	14,00		
14						

Figura 5 – Armazenamento das Coordenadas dos DOIS Pontos Conhecidos

As células de fundo amarelo são aquelas nas quais as coordenadas dos pontos

conhecidos precisam ser digitadas. Todas as demais células são calculadas pela planilha (fundo bege) ou são fixas (demais células).

A formatação da planilha aqui desenvolvida não precisa ser necessariamente seguida, mas ela já está sendo montada de forma a receber mais pontos à medida em que a complexidade aumenta.

Nas células de fundo bege serão colocados resultados da subtração das linhas versus colunas dos valores de x . Por exemplo, a célula em destaque terá o valor de $(x_1 - x_0)$.

8									
9		Valores de Y	Valores de X						
10	0	4,00			X0	1			
11	1	13,00			5,00	14,00			
12			0	5,00	-	=G11-E12			
13			1	14,00	-	9,00			
14									

Figura 6 – Diferenças entre as abscissas dos pontos conhecidos, duas a duas, calculadas em células da Matriz de Valores de X

Observa-se que as células da diagonal principal da Matriz de Valores de X (células de fundo bege) ficarão sempre zeradas, pois serão $(x_0 - x_0)$ e $(x_1 - x_1)$.

Necessário também definir um local para que seja possível informar a abscissa do ponto desconhecido e, logo abaixo, uma célula onde aparecerá o valor da ordenada calculada (fundo azul escuro).

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							
3		Valor calculado da ordenada do ponto procurado:							

Figura 7 – Células para as Coordenadas do Ponto Desconhecido

Finalmente, são definidas as células onde ficarão os resultados dos cálculos dos L_i . Como haverá tantos L_i quanto pontos, os L_i serão colocados na horizontal. Abaixo, é mostrada a planilha com a fórmula de L_0 já preenchida:

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							
3		Valor calculado da ordenada do ponto procurado:							
4									
5						Valores de Li			
6						0	1		
7						=(I2-G11)/(F13)			
8									
9		Valores de Y		Valores de X					
10	0	4,00				X0	1		
11	1	13,00				5,00	14,00		
12				0	5,00	-	9,00		
13				1	14,00	-	9,00		
14									

Figura 8 – Vetor Linha onde serão calculados $L_0(x)$ e $L_1(x)$

Células usadas no cálculo de $L_0(x)$:

- I2 = Onde é informada a abscissa x do ponto desconhecido, cuja ordenada y se deseja calcular;
- G11 = Onde estará o valor de X_1 ;
- F13 = Célula calculada como sendo $(x_0 - x_1)$;
- $F7 = \frac{(I2-G11)}{F13} = \frac{(x-x_1)}{(x_0-x_1)} = L_0$;
- Analogamente, $G7 = L_1$.

A Célula I3 receberá a fórmula para o cálculo da ordenada procurada usando os valores de L_0 e L_1 calculados em F7 e G7 respectivamente.

Informe o valor da abscissa do ponto a ter sua ordenada calculada:							
Valor calculado da ordenada do ponto procurado:							=B10*F7+B11*G7

Figura 9 – Célula onde aparecerá o Valor da Ordenada do Ponto Desconhecido

Células usadas no cálculo de y:

- B10 = y_0 e B11 = y_1 ;
- $F7 = L_0$ e $G7 = L_1$ (conforme já visto na Figura 8);

- $I3 = B10 * F7 + B11 * G7 = y_0L_0 + y_1L_1 = p(x) = y.$

Após a realização de todos esses passos, a planilha está pronta. Necessário, agora, testá-la.

Teste 1: informar a abscissa do primeiro Ponto Conhecido. Se for colocado em I2 o valor de x_0 , a célula I3 tem que retornar y_0 .

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							5
3		Valor calculado da ordenada do ponto procurado:							4
4									
5						Valores de L_i			
6						0	1		
7						1,00	-		
8									
9		Valores de Y		Valores de X					
10	0	4,00				X0	1		
11	1	13,00				5,00	14,00		
12				0	5,00	-	9,00		
13				1	14,00	-	9,00		
14									

Figura 10 – Teste 1 - Usando a Abscissa do Primeiro Ponto Conhecido

É importante notar que quando a abscissa do primeiro ponto conhecido (x_0) é informada na célula I2, a planilha encontra o valor 1 para L_0 (célula F7) e o valor 0 para L_1 (célula G7), o que se mostra totalmente coerente com o que foi visto na Equação 2.6.

Tendo sido encontrada a ordenada esperada o Teste 1 pode ser considerado como bem sucedido.

Teste 2: informar a abscissa do segundo Ponto Conhecido. Se for colocado em I2 o valor de x_1 , a célula I3 tem que retornar y_1 .

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							14
3		Valor calculado da ordenada do ponto procurado:							13
4									
5						Valores de L_i			
6						0	1		
7						-	1,00		
8									
9		Valores de Y		Valores de X					
10	0	4,00				X0	1		
11	1	13,00				5,00	14,00		
12				0	5,00	-	9,00		
13				1	14,00	-	9,00	-	
14									

Figura 11 – Teste 2 - Usando a Abscissa do Segundo Ponto Conhecido

Analogamente ao Teste 1, é possível notar que, desta vez, quando a abscissa do segundo ponto conhecido (x_1) é informado na célula I2, a planilha encontra o valor 0 para L_0 (célula F7) e o valor 1 para L_1 (célula G7), como era esperado que ocorresse.

Tendo sido encontrada a ordenada esperada o Teste 2 pode ser considerado como bem sucedido.

Teste 3: informar a abscissa do Ponto Desconhecido. Se for colocado em I2 o valor da abscissa do ponto desconhecido ($x = 10$), a célula I3 tem que retornar o valor da ordenada calculada na seção 4.1 ($y = 9$).

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							10
3		Valor calculado da ordenada do ponto procurado:							9
4									
5						Valores de Li			
6						0	1		
7						0,44	0,56		
8									
9		Valores de Y		Valores de X					
10	0	4,00				X0	1		
11	1	13,00				5,00	14,00		
12				0	5,00	-	9,00		
13				1	14,00	-	9,00	-	
14									

Figura 12 – Teste 3 - Usando a Abscissa do Ponto Desconhecido

Tendo sido encontrada a ordenada esperada o Teste 3 pode ser considerado como bem sucedido.

Agora o cenário a ser tratado será aquele em que há **três** pontos distintos conhecidos.

b) Há **TRÊS** pontos distintos conhecidos, e será adotada a forma direta:

Não será necessário construir uma nova planilha para este novo cenário, bastando fazer algumas adaptações na planilha feita para o item anterior. Como agora há **três** valores para y e **três** valores para x, a estrutura já criada deve ser ampliada para que passe a conter as coordenadas do terceiro ponto. Usando os valores do item (b) da seção 4.1 a planilha ficará assim:

8									
9		Valores de Y		Valores de X					
10	0	1,25				0	1	2	
11	1	-	0,75			0,50	1,50	4,00	
12	2	3,00		0	0,50	-	1,00	3,50	
13				1	1,50	-	1,00	-	2,50
14				2	4,00	-	3,50	-	2,50
15									

Figura 13 – Armazenamento das Coordenadas dos TRÊS Pontos Conhecidos

A Matriz das Diferenças também já foi atualizada. Destaque para a Diagonal Principal que, como já comentado anteriormente, deve sempre conter zeros.

O Vetor Linha para cálculo e armazenamento dos $L_i(x)$ também foi ampliado para conter mais uma coluna, para o $L_2(x)$.

Valores de L_i		
0	1	2
1,00	-	-

Figura 14 – Vetor Linha ampliado para conter $L_2(x)$

Agora, a planilha já se encontra adaptada e será preciso testá-la usando os três pontos já conhecidos e o ponto a ser calculado.

Teste 1: informar a abscissa do primeiro Ponto Conhecido. Se for colocado em I2 o valor de x_0 , a célula I3 tem que retornar y_0 .

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							0,5
3		Valor calculado da ordenada do ponto procurado:							1,25
4									
5						Valores de L_i			
6						0	1	2	
7						1,00	-	-	
8									
9		Valores de Y		Valores de X					
10	0	1,25				0	1	2	
11	1	-	0,75			0,50	1,50	4,00	
12	2		3,00	0	0,50	-	1,00	3,50	
13				1	1,50	-	1,00	-	2,50
14				2	4,00	-	3,50	-	2,50
15									

Figura 15 – Teste 1 - Usando a Abscissa do Primeiro Ponto Conhecido

Mais uma vez, deve-se observar que quando a abscissa do primeiro ponto conhecido (x_0) é informada na célula I2, a planilha encontra o valor 1 para L_0 (célula F7) e o valor 0 para L_1 e L_2 (células G7 e H7, respectivamente).

Tendo sido encontrada a ordenada esperada o Teste 1 pode ser considerado como bem sucedido.

Teste 2: informar a abscissa do segundo Ponto Conhecido. Se for colocado em I2 o valor de x_1 , a célula I3 tem que retornar y_1 .

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							1,5
3		Valor calculado da ordenada do ponto procurado:							-0,75
4									
5						Valores de L_i			
6						0	1	2	
7						-	1,00	-	
8									
9		Valores de Y		Valores de X					
10	0	1,25				0	1	2	
11	1	-	0,75			0,50	1,50	4,00	
12	2		3,00	0	0,50	-	1,00	3,50	
13				1	1,50	-	1,00	-	2,50
14				2	4,00	-	3,50	-	2,50

Figura 16 – Teste 2 - Usando a Abscissa do Segundo Ponto Conhecido

Dessa vez, como esperado, a planilha encontra o valor 1 para L_1 (célula G7) e o valor 0 para L_0 e L_2 (células F7 e H7, respectivamente).

Tendo sido encontrada a ordenada esperada o Teste 2 pode ser considerado como bem sucedido.

Teste 3: informar a abscissa do terceiro Ponto Conhecido. Se for colocado em I2 o valor de x_2 , a célula I3 tem que retornar y_2 .

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							4
3		Valor calculado da ordenada do ponto procurado:							3
4									
5						Valores de Li			
6						0	1	2	
7						-	-	1,00	
8									
9		Valores de Y		Valores de X					
10	0	1,25				0	1	2	
11	1	-	0,75			0,50	1,50	4,00	
12	2		3,00	0	0,50	-	1,00	3,50	
13				1	1,50	-	1,00	-	2,50
14				2	4,00	-	3,50	-	2,50

Figura 17 – Teste 3 - Usando a Abscissa do Terceiro Ponto Conhecido

Agora, o valor de $L_0=L_1=0$ e o valor de $L_2=1$.

Tendo sido encontrada a ordenada esperada o Teste 3 pode ser considerado como bem sucedido.

Teste 4: informar a abscissa do Ponto Desconhecido. Se for colocado em I2 o valor da abscissa do ponto desconhecido ($x = 3$), a célula I3 tem que retornar o valor da ordenada calculada na seção 4.1 ($y = 0$).

	A	B	C	D	E	F	G	H	I
1									
2		Informe o valor da abscissa do ponto a ter sua ordenada calculada:							3
3		Valor calculado da ordenada do ponto procurado:							0
4									
5						Valores de Li			
6						0	1	2	
7						-	0,43	1,00	0,43
8									
9		Valores de Y		Valores de X					
10	0	1,25				0	1	2	
11	1	-	0,75			0,50	1,50	4,00	
12	2		3,00	0	0,50	-	1,00	3,50	
13				1	1,50	-	1,00	-	2,50
14				2	4,00	-	3,50	-	2,50

Figura 18 – Teste 4 - Usando a Abscissa do Ponto Desconhecido

Tendo sido encontrada a ordenada esperada o Teste 4 pode ser considerado como bem sucedido.

Após ser demonstrada a forma de adaptação da planilha **do** caso (a) onde havia apenas dois pontos conhecidos, **para** o caso (b) com três pontos conhecidos, é possível perceber que é fácil adaptar a planilha aqui construída para cenários com mais de três pontos conhecidos; para isso, basta realizar os seguintes procedimentos:

- 1) Ampliar a matriz de Valores de x aumentando a quantidade de linhas e de colunas até atingir a quantidade de pontos distintos conhecidos; preencher os valores de cada x nos cabeçalhos horizontal e vertical da matriz (**linha 11** e **coluna E**, respectivamente); e, por fim, ajustar os cálculos das células da matriz. Importante lembrar que a Diagonal Principal dessa Matriz tem que ficar sempre zerada;
- 2) Ampliar o vetor-coluna de valores de y , aumentando a quantidade de linhas até atingir a quantidade de pontos distintos conhecidos e preencher o vetor com os valores de cada y ;
- 3) Ampliar o vetor-linha de valores de L_i , aumentando a quantidade de colunas até atingir a quantidade de pontos distintos conhecidos e atualizar as fórmulas da linha 7 para que cada L_i seja adequadamente calculado²⁰.

5.2 A Fórmula Geral Escrita em Formato Iterativo

À primeira vista, pode-se pensar que para desenvolver um programa de computador que faça todos os cálculos feitos pela planilha eletrônica é necessário construir uma estrutura idêntica àquela construída na planilha, mas não é bem assim que se faz.

Evidentemente a lógica aplicada a um programa tem que seguir o Método de Lagrange e produzir basicamente os mesmos resultados da planilha, mas a forma de construção é um tanto diferente.

Idealmente, um programa deve ter uma lógica otimizada e iterativa, o que significa que não se pode calcular cada elemento da planilha de forma isolada, ocupando uma linha de código para cada elemento. É necessário buscar uma forma de criar laços de repetição em que, a cada ciclo, um elemento seja calculado. Essa técnica torna o programa mais enxuto, com menos linhas.

Para que tal abordagem seja possível é preciso identificar padrões de repetição que o método a ser programado possui e, a partir desses padrões, programar os comandos a serem colocados dentro do laço a ser repetido.

²⁰ Sugestão: experimente forçar o valor 1 para a diagonal principal da matriz de Valores de x e veja como o cálculo de L_i se torna um pouco mais fácil de ser feito, pois não precisaria eliminar o fator zero do denominador, pois ele se tornaria 1, valor este que, por ser o Elemento Neutro da Multiplicação, não afeta o resultado.

O raciocínio aqui seguido terá como ponto de partida a equação 2.2, mostrada na seção 2.3:

$$p_n(x_i) = y_0 L_0(x_i) + y_1 L_1(x_i) + \dots + y_n L_n(x_i) = y_i.$$

É evidente o fato de que $p_n(x)$ é calculado através de um somatório, onde cada parcela apresenta um índice para y que é o mesmo índice de L .

Então, essa equação pode ser reescrita na forma de um somatório:

$$p_n(x_i) = \sum_{k=0}^n y_k L_k(x_i). \tag{5.1}$$

É possível perceber que o *somatório é um laço de repetição*, onde o(s) índice(s) varia(m) para determinação de cada parcela a ser somada. Neste caso há apenas um índice variando (k) de 0 a n .

Agora, observando-se as equações 2.10, 2.11 e 2.12 mostradas na seção 2.3:

$$L_0(x) = \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)}$$

$$L_1(x) = \frac{(x-x_0)(x-x_2)\dots(x-x_n)}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_n)}$$

.....

$$L_n(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_{n-1})}{(x_n-x_0)(x_n-x_1)\dots(x_n-x_{n-1})}.$$

É evidente o fato de que $L_n(x)$ é calculado através de uma razão de dois produtórios. No numerador aparecem todos os elementos $(x - x_k)$, desde que o índice k não seja igual ao índice de $L(x)$. Já no denominador, cada fator apresenta o x de mesmo índice de $L(x)$ e, à semelhança do numerador, estão presentes todos os elementos $(x_n - x_k)$, desde que o índice k não seja igual ao índice de $L(x)$. Portanto, como o comportamento dos índices do produtório do numerador é bastante semelhante ao do produtório do denominador, é possível considerá-lo como um único produtório ao invés de tratá-lo como uma razão de produtórios. Assim, resulta a seguinte expressão para $L_n(x)$:

$$L_k(x) = \prod_{j=0, j \neq k}^n \frac{(x - x_j)}{(x_k - x_j)}. \tag{5.2}$$

Da mesma forma como afirmado para o *somatório*, o *produtório também é um laço de repetição*, onde o(s) índice(s) varia(m) para determinação de cada fator a ser multiplicado.

Neste caso há dois índices (k e j) sendo que k é o mesmo índice usado no somatório e j é um índice exclusivo do produtório. Além disso, existe um critério de exclusão, pois j tem que ser diferente de k . Este critério de exclusão determina que, à medida em que os índices vão variando, sempre que j for igual a k , este fator deve ser desconsiderado no cálculo do produto final. Um outro ponto interessante de ser destacado é que, como o produtório tem como um dos índices aquele já usado no somatório, fica claro que o *laço de repetição do somatório* será o “laço de fora” e o *laço de repetição do produtório* será o “laço de dentro”. Ou seja, é necessário primeiro variar o índice k e, para cada k , variar o índice j .

Acoplando a Equação 5.2 para dentro da Equação 5.1, surge uma equação mais completa:

$$p_n(x_i) = \sum_{k=0}^n y_k \left(\prod_{j=0, j \neq k}^n \frac{(x - x_j)}{(x_k - x_j)} \right). \quad (5.3)$$

Surge, assim, um jeito iterativo de calcular, pela **Forma Direta**, a ordenada de um ponto desconhecido a partir de um conjunto de pontos distintos conhecidos, usando o Método de Interpolação Polinomial de Lagrange.

5.3 O Algoritmo que Implementa os Laços de Repetição

A partir da identificação de como implementar o método de forma iterativa usando estruturas que representam laços de repetição, é possível traduzir estas estruturas em comandos de computador para que a máquina possa fazer os cálculos sempre que for necessário.

Para tanto é necessário construir um programa que executará esses laços, mas, inicialmente, e por questões didáticas, será escrito primeiro um algoritmo em português que, depois, será traduzido para uma linguagem de programação.

Um algoritmo é, basicamente, uma estrutura lógica que define as tarefas que precisam ser realizadas (por um ser humano, manualmente, ou pelo programa de computador) e a ordem em que serão realizadas, para que o resultado desejado seja produzido. O algoritmo a ser desenvolvido aqui terá os seguintes passos (observe o quanto a planilha eletrônica montada anteriormente servirá de guia na montagem do algoritmo):

- Passo 1 - Definir quantos pontos distintos são conhecidos. Isto é, definir quanto será $n + 1$. Tendo o valor de $n + 1$, evidentemente será conhecido o valor de n ;
- Passo 2 – Definir as coordenadas destes $n + 1$ pontos conhecidos e armazená-las em dois vetores, um para as abscissas e outro para as ordenadas, cujo primeiro valor estará na ocorrência 0, o segundo na ocorrência 1, e assim sucessivamente;

- Passo 3 – Receber a abscissa do ponto desconhecido;
- Passo 4 – Abrir um laço de repetição que vai variar o índice k do somatório (o laço de fora citado acima), desde 0 até n ;
- Passo 5 – Dentro do primeiro laço, abrir um segundo laço de repetição (o laço de dentro que citado acima) que vai variar o índice do produtório, chamado de j , desde 0 até n ;
- Passo 6 – Para cada ciclo do laço de dentro, se os índices j e k forem diferentes, calcular $(x - x_j)$ e $(x_k - x_j)$ e a razão entre eles, multiplicando essa razão pelo produto parcial que estará acumulado até a iteração anterior e ir para o próximo ciclo; caso i e j sejam iguais, não efetuar cálculo algum e ir para o próximo ciclo;
- Passo 7 – Após o esgotamento dos ciclos do laço de dentro, multiplicar o produto final calculado pela variável y do índice k , somar este produto final à soma parcial que estará acumulada até a iteração anterior, e ir para o próximo ciclo do laço de fora;
- Passo 8 – Após o esgotamento dos ciclos do laço de fora, o valor de $P_n(x)$, que é o valor da ordenada do ponto desconhecido de abscissa x , estará calculado e poderá ser mostrado.

Abaixo é mostrado tal algoritmo desenvolvido com base nos passos acima descritos:

```

- (00) **** Início do Algoritmo ***
- (01) Informar quantos pontos distintos são conhecidos -> Qtd
- (01) Calcular n = Qtd - 1
- (02) Informar as coordenadas de todos os pontos distintos conhecidos
- (02) Armazenar no VetorX as Abscissas dos pontos conhecidos (serão n + 1 ocorrências, de 0 a n)
- (02) Armazenar no VetorY as Ordenadas dos pontos conhecidos (serão n + 1 ocorrências, de 0 a n)
- (03) Informar a Abscissa do ponto desconhecido -> X
- (03) A Ordenada será calculada na variável Y, que deverá, no início do programa, receber o valor 0
- (04) IndiceK recebe o valor 0
- (04) Enquanto IndiceK for menor ou igual a n, faça:
- (05)   Variável L recebe 1
- (05)   IndiceJ recebe o valor 0
- (05)   Enquanto IndiceJ for menor ou igual a n, faça:
- (06)     Se IndiceK diferente de Indicej
- (06)       Calcular Numerador = ( X - VetorX[ocorrência = IndiceK] )
- (06)       Calcular Denominador = ( VetorX[ocorrência = IndiceK] - VetorX[ocorrência = IndiceJ] )
- (06)       Calcular L = L * Numerador / Denominador
- (06)     Fim do Se
- (06)     Ir para o próximo IndiceJ
- (06)   Fim do Laço de dentro
- (07)   Calcular Y = Y + VetorY[ocorrência = IndiceK] * L
- (07)   Ir para o próximo IndiceK
- (07) Fim do laço de fora
- (08) Mostrar o valor de Y (que é a ordenada do ponto desconhecido)
- (99) **** Fim do Algoritmo ***

```

Figura 19 – Algoritmo em Linguagem Natural para Implementação da Forma Direta

Algumas **observações** importantes a respeito do algoritmo apresentado, para uma melhor compreensão:

- Entre parênteses, à esquerda de cada linha, foi colocada a indicação do passo (dos descritos acima) ao qual cada linha se refere;
- Para que seja possível focar no que é realmente importante na lógica proposta, não estão sendo considerados os passos para fazer as iterações com o usuário para montagem dos vetores X e Y . Quando for montado o programa esses vetores já estarão pré-preenchidos, precisando que o seu conteúdo seja alterado a cada novo cenário a ser validado;
- É importante observar que esta lógica independe de quantos são os pontos distintos conhecidos, uma vez que os laços vão de 0 a n , sendo n obtido no passo 01;
- As linhas dos laços (representados pela palavra “Enquanto”) vão variando o Índice até que seja ‘menor ou igual a n ’. O mesmo efeito seria obtido se fosse ‘menor que $n + 1$ ’.

Como o algoritmo não está escrito em português não estruturado, ele não pode ser considerado como um programa de computador. Mas, exatamente por ter sido escrito em português, ele facilita o entendimento da estrutura lógica que precisa ser seguida para desempenhar a tarefa e obter o resultado desejado.

A próxima tarefa é transcrever esse algoritmo do português para uma linguagem que possa ser interpretada pelo computador.

5.4 O Programa que Implementa os Laços de Repetição

A ‘tradução’ de um algoritmo em português para um programa de computador é uma tarefa tão simples quanto a clareza e abrangência do algoritmo. Neste caso aqui, a implementação será simples, porque o algoritmo está tão detalhado que, praticamente, apenas uma tradução quase literal será necessária.

A linguagem utilizada neste trabalho é C# (lê-se CSharp). É preciso que se diga que a escolha da linguagem a ser utilizada depende muito do gosto e conhecimento pessoal do programador e das ferramentas disponíveis no seu ambiente de desenvolvimento.

Como o foco deste trabalho não é ensinar ao leitor as técnicas de programação ou mesmo os comandos da linguagem escolhida e sua sintaxe, não será produzido um passo a passo da construção do programa, pois isto já está suficientemente explicado no algoritmo. Mas para uma melhor compreensão da lógica desenvolvida, à direita de cada comando foi colocado um comentário “/* . . . */” mostrando a qual passo do algoritmo aquela linha de código corresponde.

Seguindo o que foi comentado na segunda observação ao final da seção anterior, o programa já estará preparado para a execução do problema apresentado na letra (a) da seção 4.1: **dois** pontos distintos conhecidos, (5 , 4) e (14 , 13). Percebe-se isso pelas

linhas de código mostradas abaixo em que é possível ver o conteúdo dos vetores VetorX e VetorY já preenchidos. Observe-se que o campo Qtd também precisa ser preenchido com a quantidade de pontos conhecidos cujas coordenadas serão preenchidas nos vetores citados:

```
int Qtd = 2;
int n = Qtd - 1;

Double[] VetorX = { 5, 14 };
Double[] VetorY = { 4, 13 };
```

Figura 20 – Vetores das Coordenadas X e Y já preenchidos com os DOIS pontos conhecidos

O programa completo está mostrado abaixo:

```
/* Algoritmo do Método de Interpolação de Lagrange */
/* Dada uma abscissa via console e vetors com as coordenadas dos pontos distintos conhecidos, calcula o ordenada */

int Qtd = 2; /* (01) Quantidade de Pontos Distintos Conhecidos */
int n = Qtd - 1; /* (01) Cálculo de n = Qtd - 1 */

Double[] VetorX = { 5, 14 }; /* (02) Armazenamento das Abscissas dos pontos conhecidos */
Double[] VetorY = { 4, 13 }; /* (02) Armazenamento das Ordenadas dos pontos conhecidos */

Console.WriteLine("Informe a abscissa do ponto desejado: "); /* (03) Pede ao usuário a Abscissa do ponto desconhecido */
string PontoChar = Console.ReadLine(); /* (03) Lê resposta do usuário em cadeia de caracteres */
Double X = Convert.ToDouble(PontoChar); /* (03) Converte para numérico a informação digitada */
Double Y = 0.00; /* (03) Criando e Inicializando a Ordenada a ser calculada */

Double Numerador = 0.00; /* Criando e Inicializando a variável Numerador */
Double Denominador = 0.00; /* Criando e Inicializando a variável Numerador */
Double L; /* Criando e Inicializando a variável L */

for (int IndiceK = 0; IndiceK <= n; IndiceK++) /* (04) Laço de Fora e (07) Define incremento 1 para IndiceK */
{
    L = 1; /* (05) Inicia a Variável L para iniciar o próx. laço de dentro */
    for (int IndiceJ = 0; IndiceJ <= n; IndiceJ++) /* (05) Laço de Dentro e (06) Define incremento 1 para IndiceJ */
    {
        if (IndiceK != IndiceJ) /* (06) Aplica Regra de Exclusão comparando K com J */
        {
            Numerador = X - VetorX[IndiceJ]; /* (06) Calcula Numerador */
            Denominador = VetorX[IndiceK] - VetorX[IndiceJ]; /* (06) Calcula Denominador */
            L *= (Numerador / Denominador); /* (06) Aplica ao Produto Parcial a nova Razão Numer/Denom */
        }
    }
    Y += VetorY[IndiceK] * L; /* (07) Aplica à Soma Parcial o Termo Y * L recém calculado */
}

Console.WriteLine("Resultado: " + Y); /* (08) Mostra o Valor da Ordenada do ponto desconhecido */
Console.ReadKey(); /* (08) Aguarda para fechar a tela de execução */
```

Figura 21 – Código-Fonte do Programa de Implementação da Forma Direta

Apesar de não ter sido implementada a lógica de interação com o usuário para carga dos vetores de coordenadas dos pontos conhecidos, a interação para que seja informada a abscissa do ponto desconhecido foi feita. Então, basta agora testar o programa, usando novamente os cenários da seção 4.1.

a) Há **DOIS** pontos distintos conhecidos, e será adotada a forma direta:

Teste 1: informar a abscissa do primeiro Ponto Conhecido. Se for informado o valor de x_0 , o programa tem que retornar o valor de y_0 .

```
Informe a abscissa do ponto desejado:
```

Figura 22 – O programa pede que o usuário informe a abscissa do ponto cuja ordenada deve ser calculada

Informando-se o valor da abscissa do primeiro Ponto Conhecido e o programa gera o resultado 4, conforme esperado:

```
Informe a abscissa do ponto desejado:  
5  
Resultado: 4
```

Figura 23 – Resultado do Teste 1 - Primeiro Ponto Conhecido

Tendo sido encontrada a ordenada esperada o Teste 1 pode ser considerado como bem sucedido.

Teste 2: informar a abscissa do segundo Ponto Conhecido. Se for informado o valor de x_1 , o programa tem que retornar o valor de y_1 .

```
Informe a abscissa do ponto desejado:  
14  
Resultado: 13
```

Figura 24 – Resultado do Teste 2 - Segundo Ponto Conhecido

Tendo sido encontrada a ordenada esperada o Teste 2 pode ser considerado como bem sucedido.

Teste 3: informar a abscissa do Ponto Desconhecido.

```
Informe a abscissa do ponto desejado:  
10  
Resultado: 9
```

Figura 25 – Resultado do Teste 3 - Ponto Desconhecido

b) Há **TRÊS** pontos distintos conhecidos, e será usada a forma direta:

Antes de executar os testes deste novo cenário, é necessário alterar o programa para que, desta vez, os vetores de coordenadas dos pontos conhecidos estejam preenchidos com as coordenadas dos três pontos distintos conhecidos, (0,5 , 1,25) , (1,5 , -0,75) e (4 , 3). Observe-se que o campo Qtd também foi modificado, dizendo ao programa que há três pontos carregados nos vetores de coordenadas:

```
int Qtd = 3;
int n = Qtd - 1;

Double[] VetorX = { 0.5, 1.5, 4 };
Double[] VetorY = { 1.25, -0.75, 3 };
```

Figura 26 – Vetores das Coordenadas X e Y já preenchidos com os TRÊS pontos conhecidos

O resto do programa permanecerá exatamente o mesmo. Portanto, se for implementado um processo interativo para que o usuário informe as coordenadas dos pontos conhecidos, sejam quantos forem, o programa funcionará, pois a lógica se apoiará nas informações iniciais dadas pelo usuário (ou, nesse caso, previamente preenchidas).

Teste 1: informar a abscissa do primeiro Ponto Conhecido. Se for informado o valor de x_0 , o programa tem que retornar o valor de y_0 .

```
Informe a abscissa do ponto desejado:
0,5
Resultado: 1,25
```

Figura 27 – Resultado do Teste 1 - Primeiro Ponto Conhecido

Tendo sido encontrada a ordenada esperada o Teste 1 pode ser considerado como bem sucedido.

Teste 2: informar a abscissa do segundo Ponto Conhecido. Se for informado o valor de x_1 , o programa tem que retornar o valor de y_1 .

```
Informe a abscissa do ponto desejado:
1,5
Resultado: -0,75
```

Figura 28 – Resultado do Teste 2 - Segundo Ponto Conhecido

Tendo sido encontrada a ordenada esperada o Teste 2 pode ser considerado como bem sucedido.

Teste 3: informar a abscissa do terceiro Ponto Conhecido. Se foi informado o valor de x_2 , o programa tem que retornar o valor de y_2 .

```
Informe a abscissa do ponto desejado:  
4  
Resultado: 3
```

Figura 29 – Resultado do Teste 3 - Terceiro Ponto Conhecido

Tendo sido encontrada a ordenada esperada o Teste 3 pode ser considerado como bem sucedido.

Teste 4: informar a abscissa do Ponto Desconhecido.

```
Informe a abscissa do ponto desejado:  
3  
Resultado: 0
```

Figura 30 – Resultado do Teste 4 - Ponto Desconhecido

6 DESENVOLVIMENTO COMPUTACIONAL DO MÉTODO DE INTERPOLAÇÃO POLINOMIAL DE LAGRANGE - FORMA INDIRETA

Como visto antes, neste trabalho foram apresentadas as formas Direta e Indireta para calcular a ordenada de um ponto desconhecido a partir de sua abscissa, usando o Método de Lagrange.

Na Forma Direta foi usada a fórmula do Método para, aplicando as coordenadas dos pontos distintos conhecidos e a abscissa do ponto desconhecido, ser obtida a ordenada procurada.

Na Forma Indireta foi usada a fórmula do Método para, aplicando as coordenadas dos pontos distintos conhecidos, ser obtido o polinômio interpolador e, a este, foi aplicada a abscissa do ponto desconhecido para ser obtida a sua ordenada. Dessa forma, ganha-se tempo quando há vários pontos desconhecidos a serem calculados uma vez que se faz a aplicação da fórmula do Método uma única vez e, depois, basta usar o polinômio quantas vezes forem necessárias.

Na Forma Indireta, portanto, o objetivo é calcular os coeficientes de cada termo em x do polinômio interpolador $p_n(x)$; entretanto, não foi apresentada uma fórmula para cálculo desses coeficientes. Logo, o primeiro passo necessário é justamente deduzir esta fórmula para, somente então, poder implementar os cálculos de cada coeficiente usando uma planilha eletrônica e um programa de computador.

6.1 Determinação de uma Fórmula para Cálculo dos Coeficientes do Polinômio Interpolador

Analisando as equações 2.10, 2.11 e 2.12 descritas na seção 2.3 percebe-se que o denominador de cada $L_n(x)$ é calculável a partir das abscissas dos pontos conhecidos (com a abscissa x_n participando de todos os fatores sendo subtraída de todas as **demais** abscissas, num total de n fatores e não $n + 1$), pois a variável x não é usada no denominador.

A partir dessa observação, analisando a equação 2.2 descrita na seção 2.3, e considerando que o numerador de cada $L_n(x)$ é um polinômio de grau n^{21} é possível reescrever a equação 2.2 da seguinte forma:

$$p_n(x) = \frac{y_0}{D_0} N_0(x) + \frac{y_1}{D_1} N_1(x) + \dots + \frac{y_n}{D_n} N_n(x). \quad (6.1)$$

Onde D_i é o denominador de $L_i(x)$ e $N_i(x)$ é o numerador de $L_i(x)$, que é um polinômio em x de grau n .

Cada $N_i(x)$ será da seguinte forma:

²¹ $L_n(x)$ tem grau **IGUAL** a n e não **MENOR OU IGUAL** a n .

$$N_i(x) = \alpha_n x^n + \alpha_{(n-1)} x^{(n-1)} + \dots + \alpha_1 x + \alpha_0. \quad (6.2)$$

Cada coeficiente α de $N_i(x)$ deve ser multiplicado pelo seu respectivo (y_i / D_i) e os coeficientes de $p_n(x)$ serão determinados pelo somatório de cada uma dessas multiplicações agrupadas em termos de mesmo grau.

Algebricamente, a ideia é a seguinte:

$$N_0(x) = \alpha_n x^n + \alpha_{(n-1)} x^{(n-1)} + \dots + \alpha_1 x + \alpha_0$$

$$N_1(x) = \beta_n x^n + \beta_{(n-1)} x^{(n-1)} + \dots + \beta_1 x + \beta_0$$

$$N_2(x) = \gamma_n x^n + \gamma_{(n-1)} x^{(n-1)} + \dots + \gamma_1 x + \gamma_0$$

.....

$$N_n(x) = \omega_n x^n + \omega_{(n-1)} x^{(n-1)} + \dots + \omega_1 x + \omega_0.$$

Então, agrupando os coeficientes pelo graus dos respectivos termos:

$$\begin{aligned} p_n(x) = & \left(\frac{y_0}{D_0} \alpha_n + \frac{y_1}{D_1} \beta_n + \dots + \frac{y_n}{D_n} \omega_n \right) x^n + \\ & + \left(\frac{y_0}{D_0} \alpha_{n-1} + \frac{y_1}{D_1} \beta_{n-1} + \dots + \frac{y_n}{D_n} \omega_{n-1} \right) x^{n-1} + \\ & + \dots + \\ & + \left(\frac{y_0}{D_0} \alpha_0 + \frac{y_1}{D_1} \beta_0 + \dots + \frac{y_n}{D_n} \omega_0 \right). \end{aligned} \quad (6.3)$$

Conclusão: se for possível definir uma forma de calcular cada um dos coeficientes de cada um dos numeradores será possível definir os coeficientes de $p_n(x)$.

Agora, fazendo uma avaliação isolada de $N_0(x)$:

$$N_0(x) = (x - x_1)(x - x_2) \dots (x - x_n).$$

$$N_0(x) = \alpha_n x^n + \alpha_{(n-1)} x^{(n-1)} + \dots + \alpha_1 x + \alpha_0.$$

Percebe-se que este polinômio é expresso pelo produto de n fatores na forma $(x - x_i)$ e, portanto, as n raízes deste polinômio são x_i , $i = 1$ a n .

É possível seguir raciocínio análogo para todos os demais numeradores $N_n(x)$, apenas tomando o cuidado de que a abscissa que ficará de fora dos fatores é justamente a abscissa de mesmo índice que o polinômio do numerador (x_n). Por exemplo, na expressão

de $N_0(x)$, não aparece x_0 , na expressão de $N_1(x)$, não aparece x_1 , na expressão de $N_2(x)$, não aparece x_2 , e assim por diante, até que na expressão de $N_n(x)$, não aparece x_n . Por isso é que o grau de $N_n(x)$ é **IGUAL** a n , pois sempre haverá um dos $n + 1$ pontos distintos conhecidos ficando de fora.

O próximo passo seria trabalhar algebricamente sobre a expressão de $N_0(x)$ efetuando as várias multiplicações distributivas com o objetivo de encontrar uma expressão completa deste polinômio e tentar achar nessa expressão um padrão que possa auxiliar a calcular cada um dos $n + 1$ coeficientes. Mas não é preciso gastar tempo fazendo isso, porque as Relações de Girard²² [13] [14] [15] já dão a resposta necessária para este problema.

Assim, a partir dessas relações, é possível afirmar:

$\alpha_n = 1$, porque, por definição do Método, todos os $L(x)$ possuem coeficiente do termo de maior grau igual a 1, para facilitar os cálculos.

$$\alpha_{n-1} = (-1)^{n-\text{grau}}(x_1 + x_2 + \dots + x_n).$$

$\alpha_{n-2} = (-1)^{n-\text{grau}}(x_1x_2 + x_1x_3 + \dots + x_{n-1}x_n)$, que é a soma dos produtos das raízes, agrupadas duas a duas.

$\alpha_{n-3} = (-1)^{n-\text{grau}}(x_1x_2x_3 + x_1x_2x_4 + \dots + x_{n-2}x_{n-1}x_n)$, que é a soma dos produtos das raízes, agrupadas três a três.

.....

$$\alpha_0 = (-1)^{n-\text{grau}}x_1x_2x_3 \dots x_n, \text{ que é o produto de todas as raízes.}$$

A partir desse raciocínio, será possível automatizar o processo.

6.2 Cálculo pela Forma Indireta Utilizando uma Planilha Eletrônica

Novamente, o desenvolvimento a ser feito começará pelo o primeiro cenário usado na seção 4.1.

a) Há **DOIS** pontos distintos conhecidos, e será usada a forma indireta:

Inicialmente, é necessário organizar a forma como os dados conhecidos serão dispostos na planilha. Não será possível partir da mesma planilha usada no Capítulo 5 tendo em vista que os cálculos a serem desenvolvidos são bem diferentes. Dessa vez, os valores de x e os valores de y serão colocados na vertical (como vetores coluna). Usando os valores do item (a) da seção 4.1 será produzido:

²² Girard, Albert. (1595 – 1632), foi um brilhante matemático francês que fez diversas e importantes contribuições para o avanço da matemática, especialmente na área da álgebra. Foi o primeiro a usar os termos sin, cos e tan para as funções trigonométricas. O resultado dos seus estudos acerca da regra de formação dos coeficientes dos termos de um polinômio a partir da soma e dos produtos das raízes é o que usaremos neste trabalho.

5
6
7
8
9

n	x	f(x)
0	5	4
1	14	13

Figura 31 – Armazenamento das Coordenadas dos DOIS Pontos Conhecidos

Seguindo o mesmo padrão das planilhas do Capítulo 5, as células de fundo amarelo são aquelas nas quais as informações devem ser inseridas. Todas as demais são calculadas pela planilha (fundo bege) ou são fixas (demais células).

A formatação da planilha que está sendo desenvolvida aqui não precisa ser necessariamente seguida, mas ela já está sendo montada de forma a receber mais pontos à medida em que aumenta a complexidade dos cenários.

Em um quadro à direita, seguindo as mesmas linhas do índice dos pontos conhecidos, será calculado cada denominador D_n e cada coeficiente (y_n / D_n) de acordo com as fórmulas vistas na seção 6.1. Neste caso em que há apenas dois pontos, $D_0 = (x_0 - x_1)$ e $D_1 = (x_1 - x_0)$.

5
6
7
8
9

n	x	f(x)
0	5	4
1	14	13

Dn	Yn / Dn
-9	0,4444444444444444
9	1,4444444444444444

Figura 32 – Inclusão de Quadro para Cálculos de D_n e (y_n / D_n)

Observe-se que muitas casas decimais estão sendo usadas para que não seja perdida precisão, mas isso fica a critério de cada um. Será visto mais tarde que uma quantidade de casas decimais muito grande pode produzir alguns “desconfortos”.

Em seguida, em um outro quadro ainda mais para direita, serão calculados os coeficientes do Numerador N_n , sempre lembrando que, por definição, o **coeficiente do termo de maior grau será sempre 1**.

Neste caso em que há apenas dois pontos, o coeficiente de grau 0 de N_0 será o próprio x_1 (com sinal trocado, pois o grau do polinômio é 1 e o grau do coeficiente que está sendo calculado é 0, logo, $(-1)^{1-0} = -1$) e o coeficiente de grau 0 de N_1 será o próprio x_0 (com sinal trocado, pelo mesmo motivo já exposto). Isso vem das **Relações de Girard**, que indica que o coeficiente do termo de grau 0 será o produto de todas as raízes e cada N_i aqui possui apenas uma raiz que é o outro x). Assim:

4					Coeficientes Nn	
5					Grau 1	Grau 0
6	n	x	f(x)	Dn	Yn / Dn	
7	0	5	4	-9	0,444444444444444	-14
8	1	14	13	9	1,444444444444444	-5

Figura 33 – Inclusão de Quadro para Cálculos dos Coeficientes de N_n

Este quadro recém incluído terá todos os coeficientes α_n e β_n .

O próximo passo é multiplicar cada um desses coeficientes α_n e β_n pelo respectivo $\frac{Y_n}{D_n}$ e isto tem que ser feito para cada índice (dado pela linha) e para cada grau em separado. Necessário perceber que o que está sendo feito aqui é construir, pouco a pouco, cada elemento que está na Equação 6.3.

Dn		Yn / Dn		Coeficientes Nn		Coeficiente de cada Termo	
-9	-	0,444444444444444		Grau 1	Grau 0	Grau 1	Grau 0
9		1,444444444444444		1	-14	-0,444444444444444	6,222222222222222
				1	-5	1,444444444444444	-7,222222222222222

Figura 34 – Inclusão de Quadro para Multiplicar os Coeficientes por (y_n / D_n)

Finalmente, o último passo é somar todos os coeficientes encontrados que se refiram ao mesmo grau. Basta, então, implementar uma soma na vertical deste último quadro inserido.

Esta linha onde ficarão os somatórios, conterá os Coeficientes Finais de $p_n(x)$ e é a resposta final que se busca com a forma indireta, uma vez que eles indicarão a expressão do polinômio intepolador.

Coeficientes Nn		Coeficiente de cada Termo	
Grau 1	Grau 0	Grau 1	Grau 0
1	-14	-0,444444444444444	6,222222222222222
1	-5	1,444444444444444	-7,222222222222222
Coeficientes Finais ==>		1	-1

Figura 35 – Linha de Somatório que Traz os Coeficientes Finais de $p_n(x)$

O resultado final, indica que o coeficiente do termo de grau 1 é 1 e o coeficiente do termo de grau 0 é -1 , o que produz $p_n(x) = x - 1$, que é a resposta já conhecida para este cenário.

O principal objetivo da forma indireta é achar a expressão do polinômio interpolador, o que foi atingido. A partir desse resultado, evidentemente é possível aplicar o valor de abscissa de um ponto desconhecido, aplicar-lhe a potência correspondente a cada grau e

multiplicar os respectivos coeficientes, gerando como resultado final, o valor da ordenada deste ponto desconhecido.

Para tanto, basta serem incluídas duas linhas, como feito na planilha do Capítulo anterior, para que seja possível ao usuário informar a abscissa do ponto desconhecido e receber a ordenada como resultado.

Neste cenário, então, o cálculo da ordenada do ponto desconhecido se dará simplesmente multiplicando sua abscissa por 1 e subtraindo a quantidade 1.

Abaixo, planilha completa já com as linhas das coordenadas do ponto desconhecido:

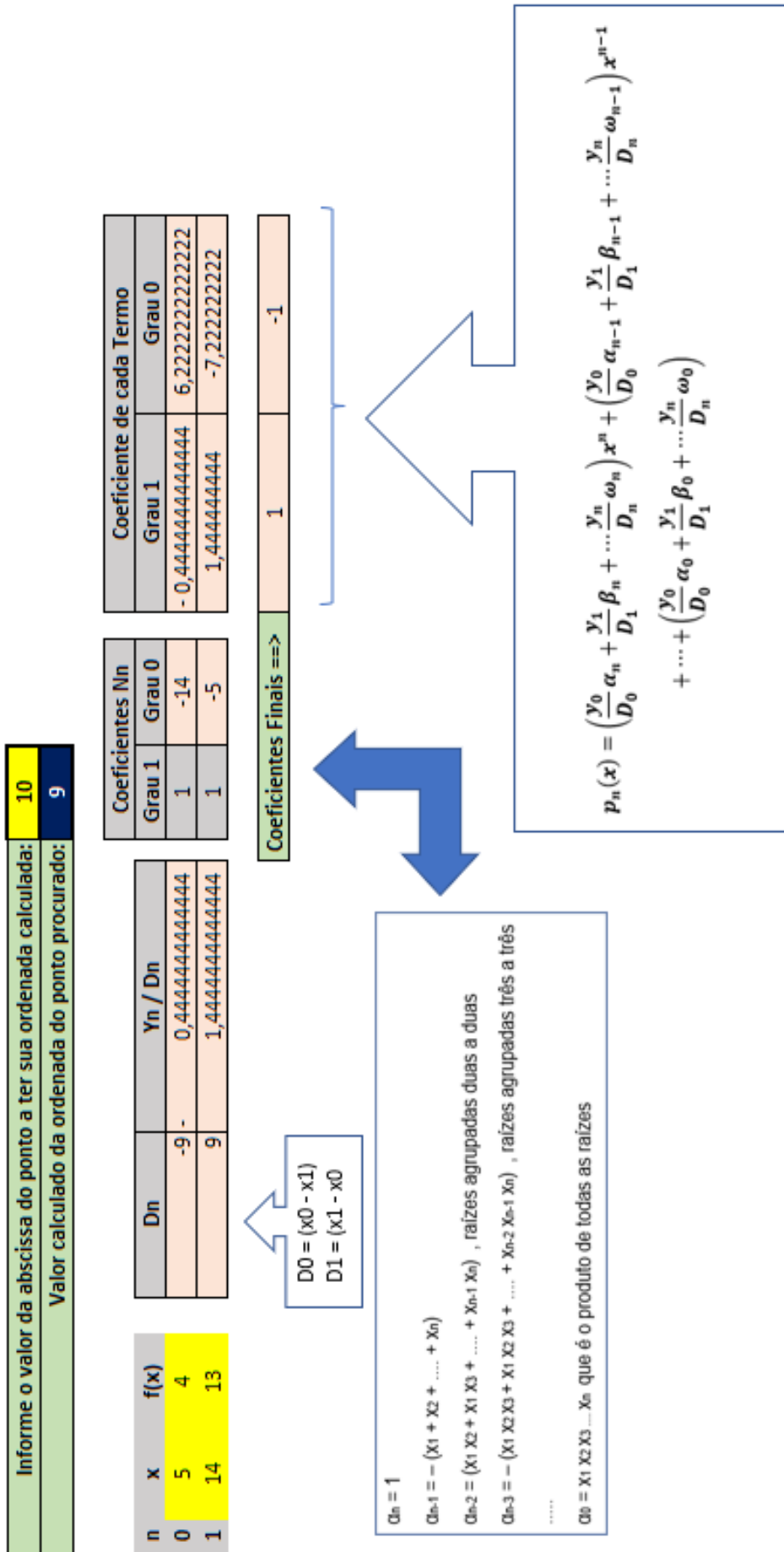


Figura 36 – Planilha Completa para Aplicação da Forma Indireta para DOIS Pontos Conhecidos

b) Há **TRÊS** pontos distintos conhecidos e será usada a forma indireta:

A planilha que acabou de ser montada para o item anterior será ligeiramente modificada para que trate um terceiro ponto conhecido. Como agora há três valores para y e três valores para x , a estrutura já criada será ampliada para que passe a conter o terceiro valor das coordenadas.

Além disso, o grau de N_n também aumentará e será necessário adaptar a planilha para esse novo cenário. Os valores do item (b) da seção 4.1 já serão colocados.

Foi adicionada uma nova linha, para o novo ponto e modificadas as fórmulas dos Denominadores, que passam a ter dois fatores cada um:

n	x	f(x)
0	0,5	1,25
1	1,5	-0,75
2	4	3

Dn	Yn / Dn
3,5	0,3571428571429
-2,5	0,3000000000000
8,75	0,3428571428571

$$D0 = (x0 - x1) * (x0 - x2)$$

$$D1 = (x1 - x0) * (x1 - x2)$$

$$D2 = (x2 - x0) * (x2 - x1)$$

Figura 37 – Planilha com Uma Linha a Mais. O denominador D passa a ter DOIS fatores.

Os coeficientes também são ajustados ganhando uma coluna a mais cada um, por conta do aumento de grau do polinômio $L_n(x)$, que passa de grau 1 para grau 2 com a adição de mais um ponto conhecido. Os cálculos também precisam ser modificados para atender as Relações de Girard. No Grau 0 fica o produto das abscissas, sempre desconsiderando a do próprio índice (por exemplo, grau 0 de N_0 será $x_1 * x_2$) e no Grau 2 fica a soma das abscissas, sempre desconsiderando a do próprio índice (por exemplo, grau 1 de N_1 será $x_0 + x_2$). Neste caso, o coeficiente do grau 0 não ficará com sinal trocado, pois como o grau do polinômio é 2 e o grau do coeficiente que está sendo calculado é 0, $(-1)^{2-0} = 1$, e o coeficiente do grau 1 ficará com o sinal trocado pelos mesmos motivos, pois $(-1)^{2-1} = -1$.

Coeficientes Nn			Coeficiente de Nn multiplicado por Yn / Dn		
Grau 2	Grau 1	Grau 0	Grau 2	Grau 1	Grau 0
1	-5,5	6	0,3571428571429	- 1,9642857142857	2,1428571428571
1	-4,5	2	0,3000000000000	- 1,3500000000000	0,6000000000000
1	-2	0,75	0,3428571428571	- 0,6857142857143	0,2571428571429
Coeficientes Finais ==>			1,0000000000000	- 4,0000000000000	3,0000000000000

Figura 38 – Matrizes de Coeficientes Modificadas para mais um Ponto Conhecido e mais um Grau no Polinômio Interpolador

Aplicados os cálculos para este cenário específico da letra (b) da seção 4.1, a linha de Coeficientes Finais indica que $p_n(x) = x^2 - 4x + 3$, que é a resposta já conhecida para este cenário.

Como já dito, o cálculo da ordenada do ponto desconhecido se dará simplesmente elevando sua abscissa ao grau indicado, multiplicando os coeficientes respectivamente e depois somando tudo.

Ainda que não seja necessário, uma vez que o principal objetivo aqui era estabelecer a expressão de $p_n(x)$, o que foi atingido com sucesso, foi feita a validação da planilha para os três pontos conhecidos e para o ponto desconhecido:

Informe o valor da abscissa do ponto a ter sua ordenada calculada:	0,5000
Valor calculado da ordenada do ponto procurado:	1,2500

Figura 39 – Teste 1 - Usando a Abscissa do Primeiro Ponto Conhecido

Informe o valor da abscissa do ponto a ter sua ordenada calculada:	1,5000
Valor calculado da ordenada do ponto procurado:	- 0,7500

Figura 40 – Teste 2 - Usando a Abscissa do Segundo Ponto Conhecido

Informe o valor da abscissa do ponto a ter sua ordenada calculada:	4,0000
Valor calculado da ordenada do ponto procurado:	3,0000

Figura 41 – Teste 3 - Usando a Abscissa do Terceiro Ponto Conhecido

Informe o valor da abscissa do ponto a ter sua ordenada calculada:	3,0000
Valor calculado da ordenada do ponto procurado:	-

Figura 42 – Teste 4 - Usando a Abscissa do Ponto Desconhecido

A seguir, a planilha completa:

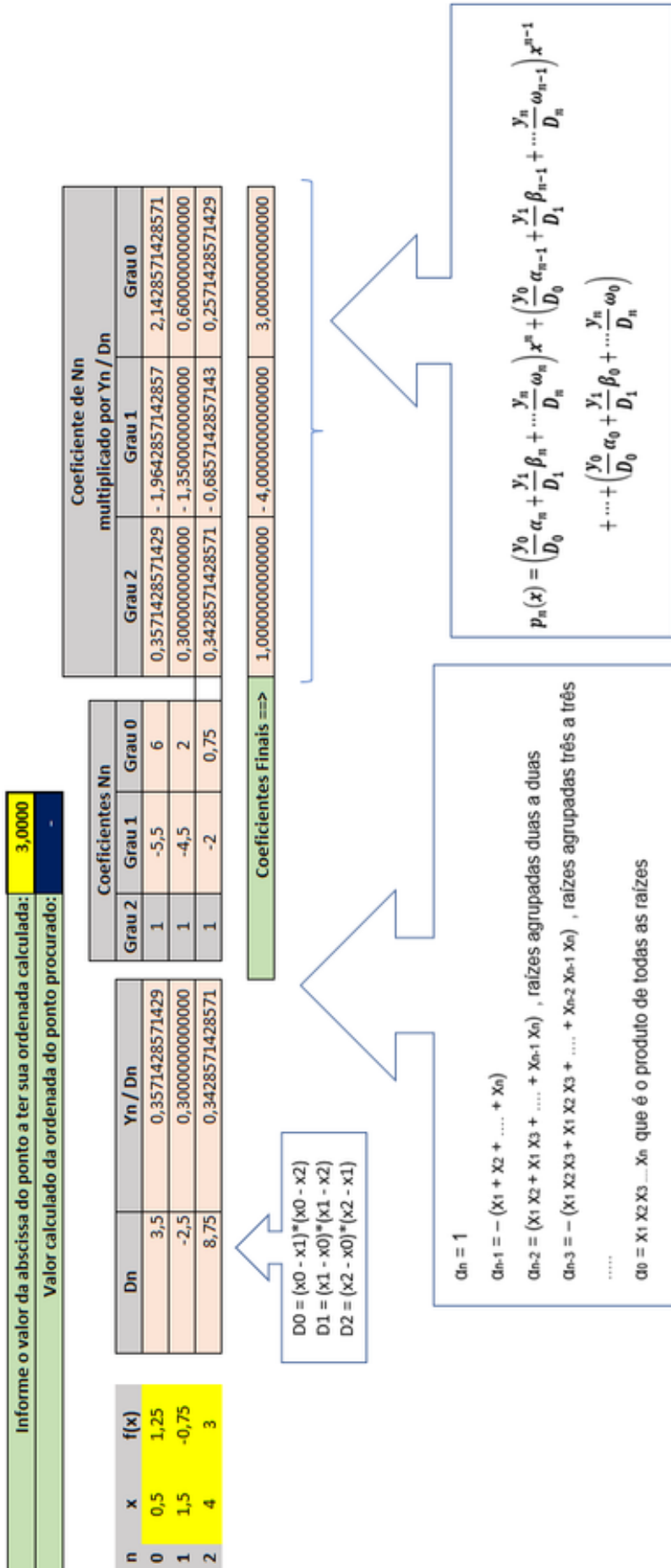


Figura 43 – Planilha Completa para Aplicação da Forma Indireta para TRÊS Pontos Conhecidos

Na sequência, para demonstrar como as coisas realmente se fecham, e que o que está sendo desenvolvido aqui é consistente, é possível verificar como essa planilha de três pontos conhecidos se comportará se nela for cadastrado aquele cenário em que há **três pontos conhecidos e colineares**. Já foi visto que o Método de Lagrange acaba anulando o termo de grau 2 e o polinômio permanece sendo uma função de grau 1.

Se a planilha estiver correta, o coeficiente de maior grau terá que ser anulado.

Preenchendo a matriz de coordenadas (fundo amarelo) com os valores dos três pontos colineares, a planilha produzirá como resultado:

Informe o valor da abscissa do ponto a ter sua ordenada calculada:		3,0000	
Valor calculado da ordenada do ponto procurado:		2,0000	

n	x	f(x)	Dn	Yn / Dn	Coeficientes Nn			Coeficiente de Nn multiplicado por Yn / Dn		
					Grau 2	Grau 1	Grau 0	Grau 2	Grau 1	Grau 0
0	5	4	45	0,0888888888889	1	-24	140	0,0888888888889	-2,1333333333333	12,4444444444444
1	14	13	36	0,3611111111111	1	-15	50	0,3611111111111	-5,4166666666667	18,0555555555556
2	10	9	-20	0,4500000000000	1	-19	70	-0,4500000000000	8,5500000000000	-31,5000000000000
Coeficientes Finais ==>					-	1,0000000000000	-1,0000000000000			

Figura 44 – Teste com TRÊS Pontos Conhecidos, porém Colineares

É importante notar que o grau 2 acabou ficando com coeficiente nulo e os coeficientes de grau 1 e 0 foram corretamente calculados, ainda resultando em $p_n(x) = x - 1$, que é a resposta já conhecida.

6.3 A Fórmula Geral Escrita em Formato Iterativo

Neste caso da Forma Indireta, enxergar a iteratividade e os laços de repetição pode ser um pouco mais difícil, pois desta vez a fórmula não dá uma indicação muito clara. Mas, usando a planilha eletrônica como guia, fica fácil perceber que cada linha segue um cálculo que se repete (com algumas adaptações); então, é possível considerar que cada linha pode ser um laço de repetição. Dentro de cada linha estão as colunas dos graus e cada grau também segue um cálculo que se repete (com algumas adaptações); então, é possível considerar que cada coluna indicativa do grau pode ser um laço de repetição.

Portanto, não há uma fórmula geral melhor do que a que já foi encontrada e é a partir desta que será construído o algoritmo da Forma Indireta.

6.4 O Algoritmo que Implementa os Laços de Repetição

A montagem de um algoritmo para a forma indireta de cálculo é uma tarefa bem mais complicada do que a montagem do algoritmo anterior. Isso porque as Relações de Girard mostram que os coeficientes são calculados por somatórios de produtos de

raízes agrupadas e cada agrupamento conterà uma quantidade diferente de elementos. Por exemplo, o coeficiente do termo de grau 0 é dado pelo produto de todas as raízes do polinômio (este é fácil) mas o coeficiente do termo de grau 1 é dado pela soma dos produtos das raízes combinadas em grupos de $n - 1$ raízes; o de grau 2 usa produtos das raízes combinadas em grupos de $n - 2$ raízes; e assim por diante.

Uma maneira mais fácil de fazer este cálculo seria colocar tantos laços de repetição, um dentro do outro, quantas forem as raízes combinadas. Entretanto, isso levaria a um programa com uma lógica não genérica.

Segue um exemplo prático das Relações de Girard para um polinômio qualquer (sem considerar aqui o sinal): se as raízes forem a, b, c e d , o grau 0 terá coeficiente $abcd$; o grau 1 terá coeficiente $(abc + abd + acd + bcd)$, com quatro parcelas, cada parcela composta por um produto de 3 fatores e quatro raízes combinadas três a três, gerando quatro combinações (daí quatro parcelas); o grau 2, terá coeficiente $(ab + ac + ad + bc + bd + cd)$; o grau 3, terá coeficiente $(a + b + c + d)$; já o grau 4, como já visto anteriormente, terá, por definição, coeficiente 1.

Para achar essas combinações no caso do grau 1 é necessário variar o primeiro fator de a até b , o segundo fator de b até c e o terceiro fator de c até d , impondo três laços, um dentro do outro. Já para o grau 2 seriam dois laços. Portanto, uma quantidade indefinida (não infinita) de raízes exigiria a existência de um algoritmo diferente para cada quantidade, o que é inviável, pois não é genérico.

Então, o objetivo aqui é encontrar um algoritmo que seja único, genérico, que contemple qualquer cenário, não importa quantos sejam os pontos conhecidos.

O algoritmo construído, e mostrado mais adiante, se baseia em quatro ideias básicas:

- 1) A **quantidade de pontos distintos conhecidos** será definida pelo usuário e, portanto, não será fixa;
- 2) A quantidade de pontos a serem agrupados a cada laço de repetição será controlada por uma variável chamada **Iteração** e a quantidade de iterações será definida pelo grau que está sendo calculado no ciclo corrente, dentro do laço;
- 3) Haverá um **vetor de controle** de dimensão $n - 1$, onde serão guardados os índices de cada X_i usado a cada iteração;
- 4) Não participa do grupo de fatores a serem usados no agrupamento o X_i da linha (ou índice) que está sendo calculado. Ou seja, no cálculo dos coeficientes da linha 0, o X_0 não será usado (e, portanto, não poderá contar como uma iteração).

Exemplo prático para tentar esclarecer: considerando-se o cenário no qual há 5 pontos conhecidos, no cálculo de cada coeficiente só poderão ser usados 4 desses 5 pontos

conhecidos (é regra do método). Inicializa-se o vetor de controle com o conteúdo [0, 1, 2, 3, 4]. Quando for calculado o coeficiente do grau 1, da linha 1, haverá grupos de 3 variáveis (regra dada pelas Relações de Girard), sendo o primeiro grupo $X_0X_2X_3$ (não será $X_0X_1X_2$, pois é necessário ignorar X_1 , uma vez que está sendo calculada a linha 1) e o último será $X_2X_3X_4$. Então, será necessário variar o primeiro fator do índice 0 até o índice 2 (pulando o 1), o segundo fator do índice 2 até o índice 3 e o terceiro fator do índice 3 até o índice 4²³.

Assim não serão necessários vários laços de repetição, dentro de vários outros laços de repetição: basta um laço onde serão evoluídos os índices das variáveis dentro do vetor de controle, sempre considerando a quantidade de iterações; esta quantidade de iterações será sempre a quantidade de pontos conhecidos, menos o grau a ser calculado, menos um (este 'menos um' é a variável a ser descartada). No exemplo prático acima, há 5 pontos conhecidos e está sendo calculado o grau 1, logo, a quantidade de iterações será $5 - 1 - 1 = 3$, que é justamente a quantidade de fatores de cada grupo a ser usado no cálculo do coeficiente daquele grau 1. Se o grau for 0, todas as variáveis serão usadas, que são 5, exceto a referente à linha sendo calculada. Então, será $5 - 0 - 1 = 4$.

No algoritmo apresentado abaixo, destaca-se:

- Passos 1 a 9 – representam a iteração com o usuário para determinar a quantidade de pontos conhecidos e suas coordenadas, a criação dos vetores onde serão armazenadas essas coordenadas, a criação do vetor de controle e a matriz onde ficarão armazenados os coeficientes à medida em que são calculados;
- Passos 10 e 11 – marcam os laços principais para cálculo dos coeficientes de cada linha e, para cada linha, o de cada grau;
- Passos 12 a 14 – define a quantidade de iterações e inicializa o vetor de controle para aquele laço;
- Passos 15 a 19 – representam o laço mais interno para cálculo de cada coeficiente da linha e grau correntes, onde é aplicada a lógica de evolução dos índices dentro do vetor de controle para definição dos agrupamentos;
- Passo 20 – armazena na matriz de coeficientes (na planilha representada pela matriz “**Coeficientes de N_n** ”) o coeficiente que acabou de ser calculado e aplica a ele a regra de sinal;
- Passo 21 – é o fim do laço principal de dentro (grau);
- Passo 22 – ao final do laço principal de dentro, o algoritmo terá chegado ao coeficiente de grau n ;

²³ De fato, os grupos serão: $X_0X_2X_3$, $X_0X_2X_4$, $X_0X_3X_4$ e $X_2X_3X_4$. Já que $\binom{4}{3} = 4$

- Passo 23 – calcula D_n . Já que ele é único para cada linha, não variando a cada grau, ele é calculado após o fim do laço principal de dentro (grau) e antes de ir para a próxima linha;
- Passo 24 – fim do laço principal de fora (linha). Quando a lógica já tiver passado por todos os graus de todas as linhas, a matriz de coeficientes estará completamente preenchida, assim como o vetor D_n ;
- Passos 25 a 31 – calcula " Y_n / D_n " para cada linha e o multiplica por cada coeficiente da matriz de coeficientes, armazenando na própria matriz o resultado de cada uma dessas multiplicações (na planilha representada pela matriz "**Coeficientes de N_n multiplicado por Y_n / D_n** ");
- Passos 32 a 37 – laços para somar os coeficientes calculados encontrando o coeficiente final de cada grau, que ficarão armazenados num vetor final; no passo 36 já calcula o valor parcial de Y (ordenada do ponto desconhecido cuja abscissa foi informada pelo usuário nos primeiros passos) à medida que o coeficiente de cada termo de $P_n(x)$ é calculado;
- Passos 38 e 39 – mostram os resultados finais, apresentando o polinômio interpolador $P_n(x)$ calculado e o valor de Y .

É importante observar que, na vida real, nos casos em que for usada a forma indireta, a execução de toda essa lógica de cálculo dos coeficientes será feita uma única vez para cada artefato ("máquina") e o polinômio calculado será armazenado em algum repositório (arquivo de dados) para que, quando houver a necessidade de calcular a ordenada de um novo ponto x desconhecido para o mesmo artefato, cujo padrão de comportamento ainda é adequadamente representado por este polinômio já calculado e armazenado, basta acessar o repositório, buscar a expressão de $p_n(x)$ e aplicar o valor de x na sua expressão para encontrar o respectivo $y = p(x)$. Em outras palavras, e como já dito anteriormente mais acima neste trabalho, se não houver mudança no conjunto de pontos distintos conhecidos, o polinômio interpolador já estará calculado e armazenado e não haverá qualquer necessidade de se refazer o seu cálculo.

Essa é uma das principais vantagens da Forma Indireta de utilização do Método de Interpolação Polinomial de Lagrange.

Abaixo, o algoritmo em linguagem natural (português não estruturado):


```

- (00) **** Início do Algoritmo ****
- (01) Informar quantos pontos distintos são conhecidos -> Qtd
- (01) Calcular IndexLim = Qtd - 1 que é o Limite do Indexador dos Laços de Repetição
- (02) Informar as coordenadas de todos os pontos distintos conhecidos
- (02) Armazenar no VetorX as Abscissas dos pontos conhecidos (serão n + 1 ocorrências, de 0 a n)
- (02) Armazenar no VetorY as Ordenadas dos pontos conhecidos (serão n + 1 ocorrências, de 0 a n)
- (03) Informar a Abscissa do ponto desconhecido -> X
- (03) A Ordenada será calculada na variável Y, que deverá, no início do programa, receber o valor 0
- (04) Criar Vetor para Armazenamento dos Cálculos Intermediários Dn, Yn/Dn
- (05) Cria Matriz dos Coeficientes dos Polinômios Intermediários
- (06) Cria Vetor para Coeficientes de Pn(x)
- (07) Cria Vetor para Controle dos Fatores para Cálculo dos Coeficientes do Grau (Girard)
- (08) Criar Diversas Variáveis para Controles dos Laços de Repetição
- (09) Cria Variáveis para Cálculos Intermediários
- (10) Índice da Linha recebe o valor 0
- (10) Enquanto Linha for menor ou igual a n (Índice Limite)
- (11) Índice do Grau recebe o valor 0
- (11) Enquanto Grau for menor que n (Índice Limite)
- (12) Cálculo de Fator de Dn (apenas se Linha for diferente de Grau)
- (13) Define quantos Fatores haverá no coeficiente do Grau atual
- (14) Inicializa Vetor de Controle para Cálculo do Coeficiente do Grau Atual
- (15) Repita até Sair, para Cálculo do Coeficiente do Grau
- (16) Repetir tantas vezes quantos fatores existir no Coeficiente do Grau
- (17) Acumular o Produto de Xi, desde que Xi não seja o da Linha atual
- (18) Ajustar o Vetor de Controle para Próxima Combinação
- (19) Se o Vetor de Controle tiver chegado ao Máximo, sair do Looping
- (19) Fim do Repita
- (20) Salva o Coeficientes do Grau calculado no Vetor de Coeficientes para a Linha atual
- (20) Considerar o sinal em função das Relações de Girard
- (21) Ir para o Próximo Grau
- (21) Fim do Laço de Dentro (Grau)
- (22) Ao final do Laço de Dentro o Grau será n, cujo coeficiente é sempre 1
- (23) Calcula Fator para Cálculo do Dn da Linha Atual e Armazena
- (24) Ir para a Próxima Linha
- (24) Fim do Laço de Fora (Linha)
- Neste Ponto todos os Dn e Coeficientes Nn estão calculados e Armazenados
- (25) Índice de Linha recebe o valor 0
- (25) Enquanto Linha for menor ou igual que n (Índice Limite)
- (26) Calcula Yn / Dn
- (27) Índice de Grau recebe o valor 0
- (27) Enquanto Grau for menor ou igual que n (Índice Limite)
- (28) Multiplica Yn / Dn pelo Coeficiente do Grau Atual da Linha Atual (salvo em (20))
- (29) Ir para o Próximo Grau
- (29) Fim do Laço de Grau
- (30) Ir para a Próxima Linha
- (31) Fim do Laço de Linha
- Neste Ponto todos os Coeficientes estão calculados para cada linha, bastando agora somá-los por grau
- (32) Índice de Grau recebe o valor 0
- (32) Enquanto Grau for menor ou igual que n (Índice Limite)
- (33) Índice de Linha recebe o valor 0
- (33) Enquanto Linha for menor ou igual que n (Índice Limite)
- (34) Acumula o Coeficiente do Grau Atual da Linha Atual
- (35) Ir para a Próxima Linha
- (35) Fim do Laço de Linha
- (36) Calcula Y parcial considerando o Grau atual e seu coeficiente final já calculado
- (37) Ir para o Próximo GRau
- (37) Fim do Laço de Grau
- Neste Ponto todos os Coeficientes Finais estão calculados para cada Grau e Y está calculado
- (38) Mostrar o Polinômio
- (39) Mostrar o Valor de Y
- (99) **** Fim do Algoritmo ****

```

Figura 45 – Algoritmo em Linguagem Natural para Implementação da Forma Indireta

As observações feitas na seção 5.3 referentes ao algoritmo da Forma Direta também são válidos para este algoritmo, portanto, é desnecessário repetí-los aqui.

6.5 O Programa que Implementa os Laços de Repetição

Abaixo, é mostrado o programa que implementa o algoritmo acima descrito para cálculo dos coeficientes do polinômio interpolador. Num primeiro momento, ele já estará

com os vetores de coordenadas preenchidos com os valores dos Pontos Conhecidos do item (a) da seção 4.1:

```

/* =====
/*                               Algoritmo do Método de Interpolação de Lagrange                               */
/* =====
/*   Dada uma Abscissa via console e vetores com as Coordenadas dos Ponto Distintos Conhecidos, calcula a Ordenada   */
/* =====
/*                               Forma Indireta -> Cálculo dos Coeficientes de Pn(x)                               */
/* =====
int Qtd = 2;                               /* (01) Quantidade de Pontos Conhecidos                               */
int IndexLim = Qtd - 1;                     /* (01) Indica o Limite do Indexador dos Laços de Repetição          */
double[] VetorX = { 5, 14 };                /* (02) Armazenamento das Abscissas dos Pontos Conhecidos          */
double[] VetorY = { 4, 13 };                /* (02) Armazenamento das Ordenadas dos Pontos Conhecidos          */

Console.WriteLine("Informe a abscissa do ponto desejado: "); /* (03) Pede ao Usuário a Abscissa do Ponto Desconhecido          */
String PontoChar = Console.ReadLine();      /* (03) Lê Resposta do Usuário em Cadeia de Caracteres             */
double X = Convert.ToDouble(PontoChar);     /* (03) Converte para Numérico a Informação Digitada               */
double Y = 0.00;                           /* (03) Cria e Inicializa a Ordenada a Ser Calculada                */

double[] VetorDn = new double[Qtd];         /* (04) Cria Vetor para Cálculos de Dn e Yn/Dn                       */
double[,] Coef = new double[Qtd, Qtd];     /* (05) Cria Matriz Coeficientes Polinômios Intermediarios          */
double[] CoeffFinal = new double[Qtd];     /* (06) Cria Vetor para Coeficientes de Pn(x)                       */
int[] VetorC = new int[Qtd];               /* (07) Cria Vetor de Controle dos Fatores (Girard)                  */

int n, j, k, Grau, Iteracao, Diferenca;    /* (08) Cria Variáveis para Controle de Laços de Repetição          */
Boolean Despreza = false;                  /* (08) Cria Variáveis para Controle de Laços de Repetição          */

Double Soma, Produto, DN;                  /* (09) Cria Variáveis para Cálculos Intermediários                 */

```

Figura 46 – Parte Inicial do Programa - Definição das Variáveis - Vetores de Coordenadas preenchidos com as Coordenadas dos DOIS Pontos Conhecidos

Como feito nos outros casos neste trabalho, o programa desenvolvido será testado aplicando-se a ele as coordenadas dos pontos já conhecidos. Os coeficientes calculados e apresentados no resultado final para o polinômio $p_n(x)$ têm que ser idênticos aos coeficientes que já são conhecidos.

```

/* *** Cálculo de Dn (no VetorDn) e dos Coeficientes Nn (na matriz Coef) (Ver Planilha) ***
for (n = 0; n <= IndexLim; n++) /* (10) Laço de Fora - Linha (n) com Incremento 1 */
{
    DN = 1; /* Inicia Variável DN com 1 (Elemento Neutro Multiplicação) */
    for (Grau = 0; Grau < IndexLim; Grau++) /* (11) Laço de Dentro - Grau com Incremento 1 */
    {
        if (Grau != n) DN *= (VetorX[n] - VetorX[Grau]); /* (12) Calcula e Acumula Fator Dn (se n diferente de Grau) */
        Iteracao = IndexLim - Grau; /* (13) Iteração = Qtd de Fatores do Coeficiente do Grau */
        for (int i = 0; i <= IndexLim; i++) VetorC[i] = i; /* (14) Inicializa Vetor de Controle com [ 0,1,2,...,n] */
        Soma = 0; /* Inicializa Variável para Cálculo do Coeficiente do Grau */
        while (true) /* (15) Laço para Cálculo do Coeficiente do Grau */
        {
            Produto = 1.00; /* Inicializa Variável para Cálculo do Coeficiente do Grau */
            Despreza = false; /* Inicializa Variável para Pular Fatores da mesma Linha */
            for (int i = 0; i <= Iteracao - 1; i++) /* (16) Laço para Cálculo da Parcela do Coeficiente do Grau */
            {
                Produto *= VetorX[VetorC[i]]; /* (17) Calcula a Parcela do Coeficiente do Grau... */
                if (VetorC[i] == n) Despreza = true; /* (17) ... desde que o Xi não pertença à linha atual */
            }
            if (!Despreza) Soma += Produto; /* (17) Acumula a Parcela do Coeficiente do Grau */
            j = Iteracao - 1; /* (18) Ajusta o Vetor de Controle para Identificação..... */
            Diferenca = 0; /* (18) ... da próxima combinação a ser usada no cálculo... */
            while (j >= 0) /* (18) ... do coeficiente do Grau pela Relação de ..... */
            {
                VetorC[j] += 1; /* (18) ... Girard em função do Grau atual */
                if (VetorC[j] > IndexLim - Diferenca) /* (18) */
                {
                    j -- 1; /* (18) */
                    Diferenca += 1; /* (18) */
                }
                else /* (18) */
                {
                    for (k = j + 1; k <= Iteracao - 1; k++) /* (18) */
                    {
                        VetorC[k] = VetorC[k - 1] + 1; /* (18) */
                        break; /* (18) */
                    }
                }
            }
            if (VetorC[0] > Qtd - Iteracao) break; /* (18) Fim do Ajuste do Vetor de Controle */
        }
        Coef[n, Grau] = Soma * Math.Pow( -1 , IndexLim - Grau); /* (20) Armazena Coeficiente do Grau calculado c/ Sinal */
    }
    Coef[n, Grau] = 1; /* (21) Vai para o Próximo Grau até que o Grau seja = n */
    if (Grau != n) DN *= (VetorX[n] - VetorX[Grau]); /* (22) Coeficiente 1 para Grau = n */
    VetorDn[n] = DN; /* (23) ... e Armazena Fator Acumulado no Vetor Apropriado */
}
/* (24) Vai para a Próxima Linha n */

```

Figura 47 – Parte Intermediária do Programa - Lógica para Cálculo dos Dn

```

/* *** Cálculo de Dn e dos Coeficientes Nn (Ver Planilha) *** */
for (int i = 0; i <= IndexLim; i++) /* (25) Laço de Fora - Linha i com Incremento de 1 */
{
    VetorDn[i] = VetorY[i] / VetorDn[i]; /* (26) Cálculo de Yn / Dn */

    for (int t = 0; t <= IndexLim; t++) /* (27) Laço de Dentro - Grau (t) com incremento de 1 */
        Coef[i, t] *= VetorDn[i]; /* (28) Calcula Coeficiente Nn Grau Atual (29) Próximo Grau */
    /* (30) Vai para a Próxima Linha */

/* *** Somatório dos Coeficientes por Grau para Cálculo dos Coeficientes Finais de Pn(x) para cada Grau *** */
/* *** Mostra o Resultado Final do Polinômio Interpolador Pn(x) e Calcula a Ordenada do Ponto X informado no início *** */

Console.WriteLine("Pn(X) = "); /* (38) Inicia a escrita em tela do Polinômio Interpolador */
for (Grau = 0; Grau <= IndexLim; Grau++) /* (32) Laço de Fora - Grau com Incremento de 1 */
{
    CoefFinal[Grau] = 0; /* Inicia Variável para Soma dos Coeficientes do Grau */
    for (n = 0; n <= IndexLim; n++) /* (33) Laço de Dentro - Linha (n) com incremento de 1 */
        CoefFinal[Grau] += Coef[n, Grau]; /* (34) Acumula Coeficiente Grau e Linha e (35) Fim do Laço */
    if (Grau != 0) Console.WriteLine(" + "); /* (38) ..... escrita em tela do Polinômio Interpolador */
    Y += CoefFinal[Grau] * Math.Pow(X, Grau); /* (36) Calcula Y parcial considerando o Grau Atual e seu... */
    /* .... coeficiente final já calculado */
    Console.WriteLine(CoefFinal[Grau] + "X^" + Grau); /* (38) ..... escrita em tela do Polinômio Interpolador */
    /* (37) Vai para o Próximo Grau */
}
Console.ReadKey();
Console.WriteLine();
Console.WriteLine();
Console.WriteLine("=====");
Console.WriteLine("==> Pn(" + X + ") = " + Y); /* (39) Escrita em tela da Ordenada do Ponto Desconhecido */
Console.WriteLine("=====");
Console.ReadKey();

```

Figura 48 – Parte Final do Programa - Lógica para Cálculo dos Coeficientes Parciais e Finais e Exposição dos Resultados em Tela

Com o programa já pronto, ele será testado, aplicando-lhe os cenários já conhecidos.

Teste 1: informar a abscissa do Primeiro Ponto Conhecido:

```

Informe a abscissa do ponto desejado:
5
Pn(X) = -1X^0 + 1X^1

=====
==> Pn(5) = 4
=====

```

Figura 49 – Resultado do Teste 1 - Primeiro Ponto Conhecido

Teste 1 bem sucedido, pois o programa retornou o polinômio e a ordenada esperados.

Teste 2: informar a abscissa do Segundo Ponto Conhecido:

```
Informe a abscissa do ponto desejado:  
14  
Pn(X) = -1X^0 + 1X^1  
  
=====  
==> Pn(14) = 13  
=====
```

Figura 50 – Resultado do Teste 2 - Segundo Ponto Conhecido

Teste 2 bem sucedido, pois o programa retornou o polinômio e a ordenada esperados.

Teste 3: informar a abscissa do Ponto desconhecido:

```
Informe a abscissa do ponto desejado:  
10  
Pn(X) = -1X^0 + 1X^1  
  
=====  
==> Pn(10) = 9  
=====
```

Figura 51 – Resultado do Teste 3 - Ponto Desconhecido

Teste 3 bem sucedido, pois o programa retornou o polinômio e a ordenada esperados.

Mantendo o corpo principal do programa e alterando somente a parte inicial, onde são preenchidos os vetores com as coordenadas dos pontos conhecidos, para os dados do item (b) da seção 4.1, o programa alterado ficará sendo:

```

/* =====
/*                               Algoritmo do Método de Interpolação de Lagrange                               */
/* =====
/* Dada uma Abscissa via console e vetores com as Coordenadas dos Pontos Distintos Conhecidos, calcula a Ordenada */
/* =====
/*                               Forma Indireta -> Cálculo dos Coeficientes de Pn(x)                               */
/* =====
int Qtd = 3;                               /* (01) Quantidade de Pontos Conhecidos                               */
int IndexLim = Qtd - 1;                     /* (01) Indica o Limite do Indexador dos Laços de Repetição          */
double[] VetorX = { 0.5, 1.5, 4 };          /* (02) Armazenamento das Abscissas dos Pontos Conhecidos          */
double[] VetorY = { 1.25, -0.75, 3 };       /* (02) Armazenamento das Ordenadas dos Pontos Conhecidos          */

Console.WriteLine("Informe a abscissa do ponto desejado: "); /* (03) Pede ao Usuário a Abscissa do Ponto Desconhecido          */
String PontoChar = Console.ReadLine();      /* (03) Lê Resposta do Usuário em Cadeia de Caracteres              */
double X = Convert.ToDouble(PontoChar);      /* (03) Converte para Numérico a Informação Digitada                */
double Y = 0.00;                            /* (03) Cria e Inicializa a Ordenada a Ser Calculada                */

double[] VetorDn = new double[Qtd];         /* (04) Cria Vetor para Cálculos de Dn e Yn/Dn                       */
double[,] Coef = new double[Qtd, Qtd];     /* (05) Cria Matriz Coeficientes Polinômios Intermediários          */
double[] CoefFinal = new double[Qtd];      /* (06) Cria Vetor para Coeficientes de Pn(x)                       */
int[] VetorC = new int[Qtd];                /* (07) Cria Vetor de Controle dos Fatores (Girard)                  */

int n, j, k, Grau, Iteracao, Diferenca;    /* (08) Cria Variáveis para Controle de Laços de Repetição          */
Boolean Despreza = false;                  /* (08) Cria Variáveis para Controle de Laços de Repetição          */

Double Soma, Produto, DN;                  /* (09) Cria Variáveis para Cálculos Intermediários                */

```

Figura 52 – Parte Inicial do Programa - Definição das Variáveis - Vetores de Coordenadas preenchidos com as Coordenadas dos TRÊS Pontos Conhecidos

Teste 1: informar a abscissa do Primeiro Ponto Conhecido:

```

Informe a abscissa do ponto desejado:
0,5
Pn(X) = 3X^0 + -4X^1 + 1X^2

=====
=> Pn(0,5) = 1,25
=====

```

Figura 53 – Resultado do Teste 1 - Primeiro Ponto Conhecido

Teste 1 bem sucedido, pois o programa retornou o polinômio e a ordenada esperados.

Teste 2: informar a abscissa do Segundo Ponto Conhecido:

```

Informe a abscissa do ponto desejado:
1,5
Pn(X) = 3X^0 + -4X^1 + 1X^2

=====
=> Pn(1,5) = -0,75
=====

```

Figura 54 – Resultado do Teste 2 - Segundo Ponto Conhecido

Teste 2 bem sucedido, pois o programa retornou o polinômio e a ordenada esperados.

Teste 3: informar a abscissa do Terceiro Ponto Conhecido:

```
Informe a abscissa do ponto desejado:
4
Pn(X) = 3X^0 + -4X^1 + 1X^2

=====
==> Pn(4) = 3
=====
```

Figura 55 – Resultado do Teste 3 - Terceiro Ponto Conhecido

Teste 3 bem sucedido, pois o programa retornou o polinômio e a ordenada esperados.

Teste 4: informar a abscissa do Ponto Desconhecido:

```
Informe a abscissa do ponto desejado:
3
Pn(X) = 3X^0 + -4X^1 + 1X^2

=====
==> Pn(3) = 0
=====
```

Figura 56 – Resultado do Teste 4 - Ponto Desconhecido

Teste 4 bem sucedido, pois o programa retornou o polinômio e a ordenada esperados.

Por último, o programa será testado aplicando-lhe os três pontos colineares usados nos testes anteriores. Mais uma vez, espera-se encontrar um polinômio de grau 1, já que são colineares, como já foi visto anteriormente.

Abaixo a parte inicial do programa com os vetores de coordenadas preenchido com os dois pontos conhecidos do item (a) da seção 4.1 e ainda as coordenadas do ponto extra (10 , 9).

```
/* ===== */
/*                               Algoritmo do Método de Interpolação de Lagrange                               */
/* ===== */
/* Dada uma Abscissa via console e vetores com as Coordenadas dos Ponto Distintos Conhecidos, calcula a Ordenada */
/* ===== */
/*                               Forma Indireta -> Cálculo dos Coeficientes de Pn(x)                               */
/* ===== */
int Qtd = 3;                               /* (01) Quantidade de Pontos Conhecidos                               */
int IndexLim = Qtd - 1;                    /* (01) Indica o Limite do Indexador dos Laços de Repetição          */
double[] VetorX = { 05 , 14 , 10 };        /* (02) Armazenamento das Abscissas dos Pontos Conhecidos          */
double[] VetorY = { 04 , 13 , 09 };        /* (02) Armazenamento das Ordenadas dos Pontos Conhecidos          */
```

Figura 57 – Vetores de Coordenadas Preenchidos com TRÊS Pontos Colineares

Executando o programa, informando uma abscissa qualquer igual a 20:

```
Informe a abscissa do ponto desejado:
20
Pn(X) = -1X^0 + 1X^1 + 0X^2

=====
==> Pn(20) = 19
=====
```

Figura 58 – Teste com TRÊS Pontos Conhecidos, porém Colineares

O coeficiente de grau 2 é **calculado** pelo programa (já que ele recebeu 3 pontos) mas acabou **resultando em zero**, conforme esperado, e o polinômio final acabou sendo o esperado: $p_n(x) = x - 1$.

6.6 Validação Final

Até aqui tanto as planilhas quanto os programas foram testados com os mesmos cenários: **DOIS** pontos distintos conhecidos, **TRÊS** pontos distintos conhecidos e não colineares, e **TRÊS** pontos distintos conhecidos e colineares.

Para concluir esse estudo, será proposta a execução de um teste final para validar tudo o que foi visto até agora, utilizando-se um raciocínio no caminho contrário do que foi seguido até o momento, qual seja, criar aleatoriamente um polinômio de um grau qualquer, identificar tantos pontos quantos necessários com base no grau do polinômio escolhido e ver se o programa construído, a partir deste conjunto de dados conhecidos, chega à expressão do polinômio original escolhido calculando corretamente os coeficientes.

Foi escolhido para esse teste o seguinte polinômio $t(x)$ de grau 4:

$$t(x) = x^4 - 2,33x^3 - 6,25x^2 + 3,5x + 1,5.$$

Se será usado um polinômio de grau 4, então é necessário formar um conjunto de **CINCO** pontos conhecidos. Escolhidos os pontos, a função foi plotada no GeoGebra onde o gráfico abaixo foi produzido, destacados os 5 pontos conhecidos escolhidos: A (-2 , 4,14), B (-1 , -4,92), C (1 , -2,58), D (2 , -19,14) e E (3 , -26,16).

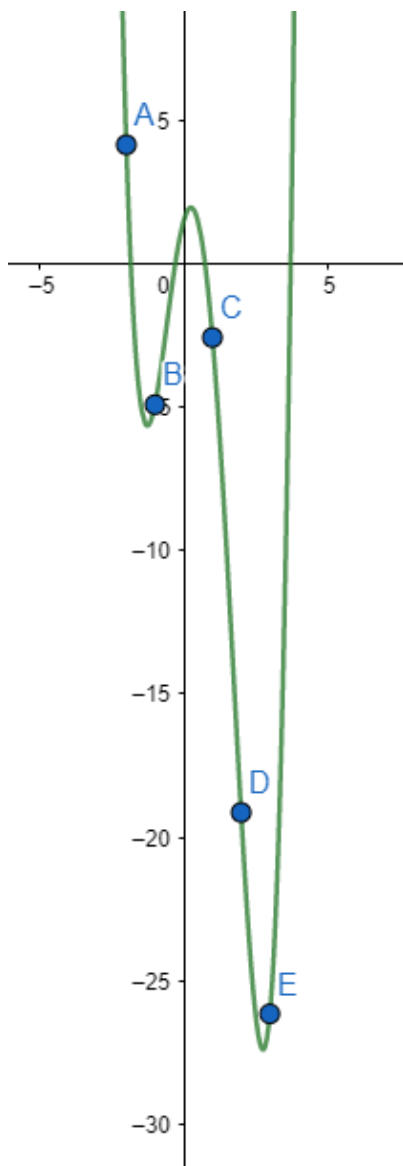


Figura 59 – $t(x)=x^4 - 2,33x^3 - 6,25x^2 + 3,5x + 1,5$ plotada via GeoGebra

Aplicando as coordenadas desses CINCO pontos conhecidos no programa, a parte inicial fica assim alterada:

```
int Qtd = 5; /* (01) Quantidade de Pontos Conhecidos */
int IndexLim = Qtd - 1; /* (01) Indica o Limite do Indexador dos Laços de Repetição */
double[] VetorX = { -2, -1, 1, 2, 3 }; /* (02) Armazenamento das Abscissas dos Pontos Conhecidos */
double[] VetorY = { 4.14, -4.92, -2.58, -19.14, -26.16 }; /* (02) Armazenamento das Ordenadas dos Pontos Conhecidos */
```

Figura 60 – Vetores de Coordenadas preenchido com os CINCO Pontos Conhecidos

E como resultado da sua execução, para um ponto desconhecido de abscissa na origem:

```
Informe a abscissa do ponto desejado:
0
Pn(X) = 1,5X^0 + 3,5X^1 + -6,25X^2 + -2,33X^3 + 1X^4
=====
==> Pn(0) = 1,5
=====
```

Figura 61 – Resultado do Teste I de Validação Final

Exatamente o mesmo polinômio escolhido ao acaso. Obtém-se desta execução as coordenadas de mais um ponto desta função: o ponto de abscissa ZERO tem ordenada 1,5.

Aplicando as coordenadas deste ponto adicional no programa:

```
int Qtd = 6; /* (01) Quantidade de Pontos Conhecidos */
int IndexLim = Qtd - 1; /* (01) Indica o Limite do Indexador dos Laços de Repetição */
double[] VektorX = { -2, -1, 1, 2, 3, 0 }; /* (02) Armazenamento das Abscissas dos Pontos Conhecidos */
double[] VektorY = { 4.14, -4.92, -2.58, -19.14, -26.16, 1.5 }; /* (02) Armazenamento das Ordenadas dos Pontos Conhecidos */
```

Figura 62 – Vetores de Coordenadas preenchido com SEIS Pontos Conhecidos

E como resultado da sua execução, com esse ponto adicional que sabe-se pertencer ao mesmo polinômio de grau 4, foi encontrado um polinômio de grau 5, conforme pode ser visto abaixo:

```
Informe a abscissa do ponto desejado:
8
Pn(X) = 1,5X^0 + 3,5X^1 + -6,25X^2 + -2,33X^3 + 1X^4 + -2,77555756156289E-17X^5
=====
==> Pn(8) = 2532,54
=====
```

Figura 63 – Resultado do Teste II de Validação Final

Mas esperava-se um polinômio de grau 4.

Entretanto, deve-se observar que o coeficiente do termo de quinto grau é $-2,7755 \times 10^{-17}$, ou seja, tão próximo a ZERO que pode ser considerado ZERO. Isto se deve a questões de precisão das variáveis do programa (conforme já comentado mais acima, isso poderia gerar algum desconforto), criadas como sendo do tipo **Double**, ou seja, precisão **Dupla**, e que armazenam uma quantidade enorme de casas decimais. Os demais coeficientes estão idênticos à execução anterior e, portanto, **este teste pode ser considerado bem-sucedido**.

Adicionalmente, vale comentar que, para os casos em que se utiliza interpolação com polinômios de ordem elevada, pode ocorrer o Fenômeno de Runge, descoberto pelo matemático alemão Carl Runge, e que propõe que polinômios de ordem elevada são, em

geral, pouco recomendáveis para a interpolação. Para maiores detalhes sobre o Fenômeno de Runge, veja [16] e [3].

7 ESTUDO ADICIONAL - O PROBLEMA INVERSO: CALCULAR A ABSCISSA A PARTIR DA ORDENADA

Até este ponto foi mostrada a utilização do Método de Interpolação Polinomial de Lagrange para identificar a ordenada de um ponto desconhecido a partir de um conjunto de $n + 1$ pontos conhecidos. Entretanto, no mundo real, nem sempre o problema que se apresenta é desta forma: a partir da abscissa, se obtém a ordenada.

Pode-se imaginar o seguinte cenário: durante o processo de testagem de uma máquina nova, o operador afere a temperatura da máquina a cada hora durante o dia inteiro e esses pares ordenados (tempo, temperatura) são considerados como sendo os $n + 1$ pontos conhecidos. Ao desenvolver o método de Lagrange, *pela forma indireta*, o engenheiro saberá qual o polinômio que melhor expressa a evolução da temperatura ao longo do tempo de uso desta máquina. Entretanto, existe um limite de segurança para a temperatura que a máquina pode alcançar a partir da qual seus componentes internos podem ser danificados. Então, o engenheiro precisa saber em que tempo (abscissa) a máquina atinge 85% da temperatura limite (ordenada) pois este é o momento em que o sistema de arrefecimento precisa ser acionado (claro que, certamente, há a possibilidade de instalar sensores térmicos que darão start no sistema de arrefecimento, mas o que se busca neste cenário proposto é a oportunidade de aplicar os estudos feitos). Portanto, ao invés de a abscissa ser conhecida e o objetivo ser calcular a respectiva ordenada, apresenta-se o problema inverso: uma ordenada para a qual se deseja conhecer a respectiva abscissa.

Inverter o par ordenado, usando a temperatura como eixo-x e o tempo como eixo-y não é possível, pois o resultado não seria uma função, uma vez que a temperatura varia ao longo do tempo e haveria mais de uma imagem para uma mesma abscissa, o que descaracterizaria tal relação como sendo uma função. Em termos matemáticos formais, o tipo de função polinomial que geralmente encontramos numa interpolação polinomial, aquelas que “sobem e descem”, não é bijetora e, portanto, não é inversível²⁴.

A solução aqui é aplicar o método de “Tentativa e Erro”, também chamado de “Método Aproximativo” ou mesmo “Algoritmo Aproximativo”. Esse método consiste basicamente em calcular a ordenada de abscissas diferentes e a cada cálculo avaliar se esta ordenada se aproxima da ordenada-alvo considerando-se uma margem de erro aceitável. Deve-se buscar reduzir gradativamente a diferença entre as abscissas à medida em que a ordenada calculada vai se aproximando da ordenada-alvo.

7.1 Método Aproximativo com uma Vizinhança Definida

²⁴ Não podemos afirmar que toda função polinomial não é inversível. Por exemplo, uma função polinomial de primeiro grau é inversível. A função $f(x) = x^3$ é polinomial e é inversível.

Quando se sabe mais ou menos em torno de qual abscissa se encontra a ordenada-alvo, este método deve ser iniciado por esta abscissa.

Exemplo de um passo a passo:

- Usa-se a abscissa X_1 como ponto de partida. Calcula-se sua ordenada como sendo Y_1 , e verifica-se que é MENOR que a ordenada-alvo;
- Soma-se à abscissa X_1 um intervalo t . Então, calcula-se a ordenada da abscissa X_2 que é $X_1 + t$, e encontra-se Y_2 ;
 - Se Y_2 for MAIOR que a ordenada-alvo, então a abscissa que atinge a ordenada-alvo estará no intervalo (X_1, X_2) . Neste caso calcula-se uma nova abscissa $X_3 = X_2 - t$ (reduzindo a abscissa anterior do incremento t), que está no intervalo (X_1, X_2) , e calcula-se sua ordenada;
 - Se Y_2 for MENOR que a ordenada-alvo e MAIOR que Y_1 , então ela está se aproximando da ordenada-alvo e, neste caso, calcula-se uma nova abscissa $X_3 = X_2 + t$, somando o incremento t , e calcula-se sua ordenada;
 - Se Y_2 for MENOR que a ordenada-alvo e MENOR que Y_1 , então ela está se afastando da ordenada-alvo e, neste caso, calcula-se uma nova abscissa $X_3 = X_1 - t$, subtraindo o incremento t , e calcula-se sua ordenada;
- Repete-se esse processo até que exatamente a ordenada-alvo seja alcançada ou até que seja alcançada uma ordenada tão próxima à ordenada-alvo quanto seja desejável (a depender da margem de erro considerada como aceitável).

Algumas considerações importantes acerca desse método:

- Uma escolha ruim para o ponto de partida pode ser muito prejudicial e levar a um esforço maior para encontrar o ponto desejado, pois há que se considerar a possibilidade de haver vários pontos que atingem a ordenada-alvo dentro de um mesmo intervalo $X_1 + t$;
- A partir do estabelecimento de um intervalo $(X_n, X_n + t)$ onde estará a ordenada-alvo, se a redução do intervalo t a cada iteração for no sentido de dividi-lo por 2, (para $t/2$), este processo é conhecido por “Pesquisa Binária” e traz uma eficiência maior ao processo, otimizando-o. Essa técnica é largamente utilizada em programação de computador;
- Por se tratar de uma função polinomial que apresenta um comportamento “sobe-e-desce” pode haver mais de uma abscissa que atinge a ordenada-alvo. Portanto, estabelecer uma única vizinhança pode deixar de fora outras abscissas que também atingiriam a ordenada-alvo.

7.2 Método Aproximativo com uma Vizinhança Definida

Quando não se sabe em torno de qual abscissa se encontra a ordenada-alvo, não há alternativa senão iniciar a pesquisa pela abscissa da extremidade do intervalo de validade do polinômio interpolador, caminhando na direção da outra extremidade, calculando-se e testando-se o comportamento das respectivas ordenadas para que seja(m) identificado(s) o(s) intervalo(s) no(s) qual(is) a ordenada-alvo está localizada. Neste caso, o passo-a-passo é basicamente o mesmo.

Vale comentar que pode-se usar a derivada primeira para encontrar pontos de máximo e mínimo locais o que pode ajudar a definir vizinhanças, mas como isso não é objeto deste trabalho, fica a cargo do leitor pesquisar, caso tenha curiosidade a respeito.

7.3 O Método de Newton

Newton criou um método aproximativo para, de forma bastante eficiente, *calcular as raízes de um polinômio*. Este método parte de um ponto x_1 que se sabe estar próximo de uma raiz e vai sucessivamente calculando o próximo ponto $x_{(n-1)}$ pela fórmula abaixo, aplicada repetitivamente:

$$x_{n+1} = x_n - \frac{p(x_n)}{p'(x_n)}$$

Onde $p'(x_n)$ é a derivada primeira de $p(x_n)$.

O limite da sequência formada por $x_1, x_2, x_3, \dots, x_n$ calculados a partir da fórmula acima é uma raiz de $p(x)$. Este algoritmo apresenta uma convergência muito rápida para a identificação da raiz do polinômio. Entretanto, vale a pena comentar que há casos em que, dependendo do ponto x_1 inicial escolhido o Método de Newton pode levar a um looping infinito que faz com que não se chegue à identificação de uma raiz válida para o polinômio estudado. Qualquer algoritmo a ser construído para implementação do Método de Newton precisa ser capaz de identificar tais situações e tratá-las quando surgirem.

Para finalizar, busca-se esclarecer abaixo duas questões que podem ter surgido para o leitor mais atento:

(1) A derivada de uma função polinomial da forma

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + \dots + a_2 x^2 + a_1 x + a_0$$

será da forma

$$p'(x) = n a_n x^{n-1} + (n-1) a_{n-1} x^{n-2} + (n-2) a_{n-2} x^{n-3} + \dots + 2 a_2 x^1 + a_1$$

sendo que a regra é muito simples: para cada termo da expressão do polinômio, o expoente original ‘desce’ multiplicando o termo e o expoente original é reduzido de 1 unidade. Como resultado disso, o grau cai de uma unidade e o termo independente do polinômio é eliminado (importante lembrar que este termo possui x com expoente zerado, e, pela regra, o 0 desce multiplicando o x_0 e, por isso, o termo naturalmente desaparece).

(2) Até então, este capítulo falava de identificar a abscissa cuja ordenada é igual a (ou muito próximo de) uma ordenada-alvo. Mas este método de Newton **serve para calcular as raízes de um polinômio** e, portanto, não tem relação com o problema original. Na verdade, tem tudo a ver. Se for chamado de Ω o valor da ordenada-alvo, e se for realizada uma translação vertical de $p(x)$ no valor de $-\Omega$, um novo polinômio $q(x)$ é obtido cujas raízes serão os pontos que atingem a ordenada Ω em $p(x)$ (veja figura abaixo, plotada pelo software GeoGebra, versão web disponível na internet):

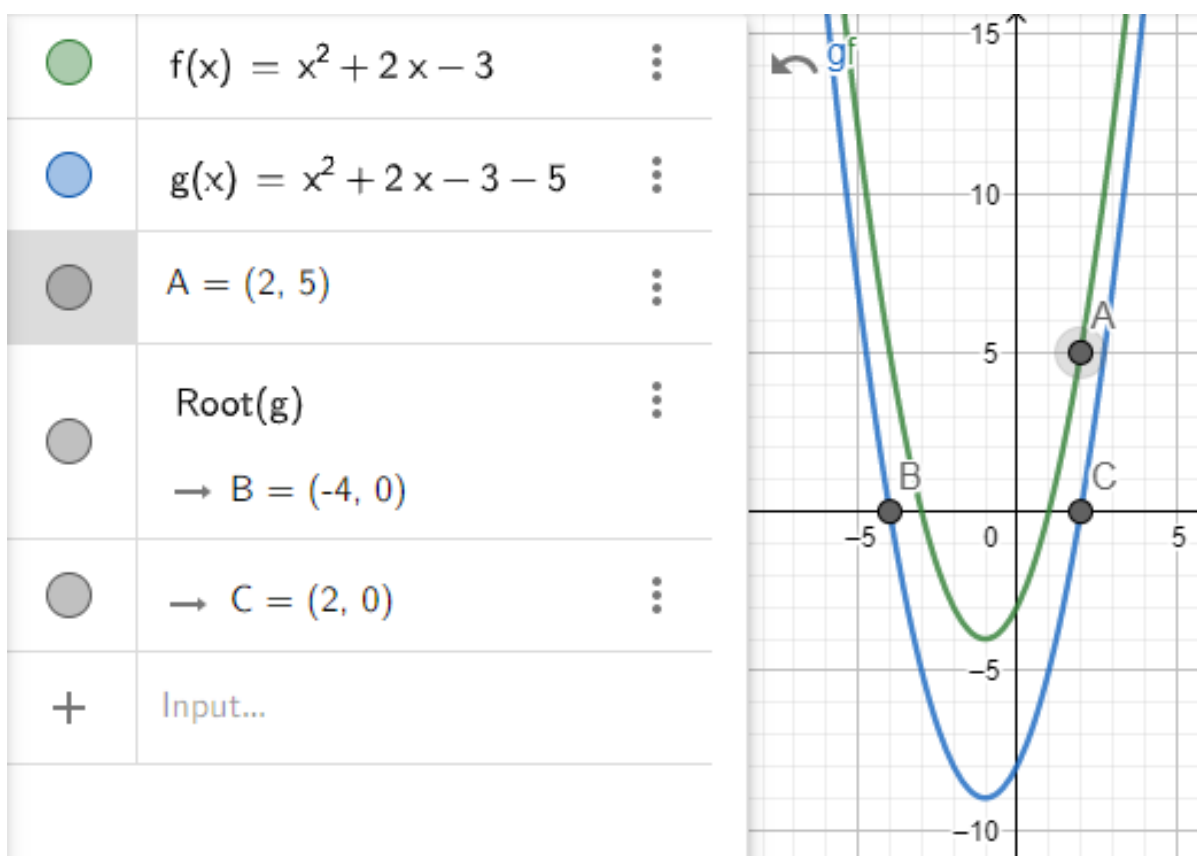


Figura 64 – Translação Vertical para uso do Método de Newton na Resolução do Problema Inverso

O ponto $A(2, 5)$ pertence à $f(x)$ (em **verde**), com ordenada 5.

Se a função $f(x)$ for transladada verticalmente para baixo em 5 unidades, será obtida uma nova função $g(x)$ (em **azul**), onde $g(x) = f(x) - 5$ que tem o ponto de abscissa 2 como uma das raízes (ponto C na figura). Portanto, com uma pequena e engenhosa adaptação, é possível usar o Método de Newton para determinação de raízes de polinômios, para a resolução do problema inverso.

8 CONSIDERAÇÕES FINAIS

A partir de um conjunto de pontos conhecidos e/ou observados, o Método de Interpolação Polinomial de Lagrange oferece uma maneira de estabelecer um padrão de comportamento, descrito através de uma função polinomial, que pode ser usado para inferir outros pontos que sejam aderentes ao padrão estabelecido. Tal Método pode ser usado de duas formas, neste trabalho batizadas de Direta e Indireta. Na Direta, é usada a fórmula geral do método e aplica-se o valor da abscissa do ponto desconhecido diretamente (daí o nome) na variável independente para obter o valor da respectiva ordenada; na Indireta, antes de aplicar o valor da abscissa do ponto desconhecido à variável independente x , desenvolve-se algebricamente a fórmula geral com o objetivo de obter o polinômio propriamente dito (ou, na verdade, os coeficientes de cada termo) para, somente então, aplicar o valor da abscissa do ponto desconhecido para obter o valor da respectiva ordenada.

Cada uma dessas duas formas possui suas vantagens e desvantagens, e foi visto que a forma direta é mais útil quando há uma variação frequente nos pontos desconhecidos e que a forma indireta é mais útil justamente quando os pontos desconhecidos não mudam e então o polinômio é calculado uma única vez para, depois, sua expressão ser usada sempre que for necessário calcular a ordenada de um ponto desconhecido.

Foram apresentadas também as Relações de Girard, ainda que de forma não profunda uma vez que não é o tema central deste trabalho mas apenas uma ferramenta extremamente útil para que os coeficientes do polinômio interpolador possam ser calculados de forma iterativa e rápida.

Foi visto também como os recursos computacionais podem ser utilizados para reduzir o tempo gasto nesses cálculos e eliminar os riscos de se obter resultados equivocados por erro humano no processo. Evidentemente, quando há poucos pontos conhecidos o risco de erro humano é menor, mas em cenários com dezenas ou mesmo centenas de pontos, a possibilidade de fazer o cálculo à mão é inimaginável e o recurso computacional é fundamental. Entretanto, muitas vezes a percepção que as pessoas de fora da área da computação têm sobre os métodos computacionais e a programação de computadores lhes dá a impressão de ser algo inatingível e que exige um esforço muito grande. Este trabalho tentou desmistificar um pouco isso, mostrando que, mesmo que o estudante não saiba programar, a planilha eletrônica, normalmente mais fácil de operar, pode servir como ferramenta muito útil para a implementação de um método computacional. De qualquer forma, a apresentação de algoritmos escritos em linguagem natural (português não estruturado) tem por objetivo facilitar a compreensão da lógica do programa desenvolvido, principalmente para os leitores pouco familiarizados com a programação de computadores.

Ao final, como sugestão de estudos adicionais, foram feitos comentários sobre a

utilidade da forma indireta de aplicação do Método de Lagrange para resolver o problema inverso, qual seja, obter a abscissa de um ponto desconhecido a partir de sua ordenada. Na ausência de uma fórmula para resolver esse problema inverso de forma objetiva e direta, foi apresentado o processo de aproximação, que consiste em escolher uma abscissa como ponto de partida e calcular sua ordenada e, a partir deste resultado, e adotando critérios adequados, escolher uma nova abscissa cuja ordenada é calculada, repetindo tal processo até que a ordenada calculada atinja a ordenada-alvo. Às vezes, não é possível atingir exatamente a ordenada-alvo e, em outras vezes, muitas iterações são necessárias para atingí-la e o processo acaba levando muito tempo; nesses casos, o usuário pode estabelecer um critério de parada baseado no quão próximo a ordenada calculada está da ordenada-alvo, quando então o laço de repetição é interrompido e a última abscissa obtida é considerada como resposta para o problema inverso. Foi, ainda, apresentado o Método de Newton, que é do tipo aproximativo, criado para determinar as raízes de um polinômio, e como ele pode ser facilmente adaptado para a solução do problema inverso de uma maneira mais eficiente, uma vez que tal método converge de forma muito mais rápida, ao ponto desejado.

Considera-se que todos os dez Objetivos Específicos descritos na sub-seção 1.3.2 foram plenamente atingidos, um a um, ao longo dos sete capítulos deste trabalho, na certeza de que, assim, o Objetivo Geral também foi atingido.

Almeja-se que este trabalho seja útil tanto ao professor de matemática quanto ao professor de informática, considerando-se aqui uma interdisciplinaridade existente entre o estudo da matemática, mais especialmente métodos de cálculo que envolvem processos repetitivos, como é o caso do Método de Lagrange, e o estudo de informática com a aplicação de planilhas eletrônicas ou mesmo de programação de computadores. Para o aluno, em especial, deseja-se que este trabalho tenha estimulado sua curiosidade em relação à possibilidade do uso da matemática para resolver problemas e, ainda, a necessidade de adentrar o mundo computacional para agilizar e otimizar o tempo despendido no processo de cálculo, deixando que a máquina faça o “trabalho braçal” para que o tempo possa ser utilizada pelo ser humano para o “trabalho intelectual”.

Como estudos futuros, foi sugerido um estudo sobre a construção de modelos computacionais que implementem o processo aproximativo usando tanto o Método de Newton quanto um Método de Pesquisa Binária, fazendo um estudo comparativo de performance e convergência. Outra frente para estudo futuro e continuidade deste trabalho seria a construção dos passos de iteração entre o usuário e o programa, que não foram codificados nos programas aqui desenvolvidos, inclusive usando interfaces padrão GUI (Graphic User Interface), acoplando ambas as abordagens (direta e indireta) em uma única solução, com possibilidade de plotagem do Polinômio Interpolador no Plano Cartesiano e dos pontos envolvidos no problema.

Referências

- 1 O'CONNOR, J. J.; ROBERTSON, E. F. **Joseph-Louis Lagrange**. 1999. Página na Internet. Disponível em: <https://mathshistory.st-andrews.ac.uk/Biographies/Lagrange/>. Acesso em: 02/03/2022. Citado na página 19.
- 2 BOYER, C. B. **História da Matemática**. 1a. ed. São Paulo: EDGARD BLÜCHER LTDA, 1974. 488 p. Citado na página 19.
- 3 GABETTA JUNIOR, A. M. **Aproximação de Funções por Interpolação: Método de Lagrange**. 2015. Dissertação (PROFMAT) — UNICAMP. Citado 2 vezes nas páginas 20 e 82.
- 4 SILVA, C. **Interpolação de Lagrange: Uma proposta ao Ensino Médio para Modelagem Matemática de Polinômios**. 2016. Dissertação (PROFMAT) — UFAM. Citado na página 20.
- 5 MARTINS, F. I. da S. **Modelagem de funções via Interpolação Polinomial de Lagrange**. 2017. Dissertação (PROFMAT) — UFG. Citado 2 vezes nas páginas 20 e 21.
- 6 SILVA, G. S. da. **Uma sequência didática para a abordagem de interpolação polinomial no Ensino Médio**. 2014. Dissertação (PROFMAT) — UFAL. Citado na página 20.
- 7 AMES, L. F. de A. **Interpolação e Aproximação de Funções por Polinômios**. 2018. Dissertação (PROFMAT) — UEM. Citado na página 20.
- 8 LAGRANGE, J. L. **Lectures on Elementary Mathematics**. 2a. ed. Chicago: The Open Court Publishing Company, 1901. 186 p. Disponível em: <https://archive.org/details/lecturesonelemen00lagruoft>. Acesso em: 05/03/2022. Citado na página 23.
- 9 RIBNIKOV, K. **Historia de Las Matematicas**. Moscou, Russia: Mir, 1987. 487 p. Citado na página 23.
- 10 IME - UNICAMP. **Os Axiomas de Euclides**. Campinas: [s.n.], 2022. Aula em Slide. Citado na página 23.
- 11 ANDRETTA, M. **Interpolação Polinomial: Diferenças Divididas de Newton**. 2012. PDF disponibilizado na Internet. Disponível em: <https://sites.icmc.usp.br/andretta/ensino/aulas/sme0500-1-12/ipnewton.pdf>. Acesso em: 05/03/2022. Citado na página 24.
- 12 LIMA, E. L. **Números e Funções Reais**. 1a. ed. Rio de Janeiro: SBM, 2017. 297 p. ISBN 978-85-85818-81-4. Citado na página 27.
- 13 O'CONNOR, J. J.; ROBERTSON, E. F. **Albert Girard**. 2010. Página na Internet. Disponível em: https://mathshistory.st-andrews.ac.uk/Biographies/Girard_Albert/. Acesso em: 11/03/2022. Citado na página 60.
- 14 VINICIUS, M. **Relações de Girard**. 2021. Vídeo-Aula. Disponível em: <https://querobolsa.com.br/enem/matematica/relacoes-de-girard>. Acesso em: 11/03/2022. Citado na página 60.
- 15 VIRTUOUS TECNOLOGIA DA INFORMACÃO. **“Polinômios” em Só Matemática**. 1998–2022. Página na Internet. Disponível em: <https://www.somatematica.com.br/emedio/polinomios/polinomios13.php>. Acesso em: 11/03/2022. Citado na página 60.

16 EPPERSON, J. F. **Fenômeno de Runge**. 2018. Disponível em: https://www.math.unipd.it/~alvise/AN_2018/LETTURE/Epperson_runge.pdf. Acesso em: 23/05/2022. Citado na página 82.