

UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL  
INSTITUTO DE MATEMÁTICA  
PROGRAMA DE PÓS GRADUAÇÃO  
MESTRADO PROFISSIONAL EM MATEMÁTICA  
EM REDE NACIONAL

LUIZ WEBERSON ALVES FELIPE

METAPOST NO ENSINO MÉDIO

CAMPO GRANDE - MS

2019

UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL  
INSTITUTO DE MATEMÁTICA  
PROGRAMA DE PÓS GRADUAÇÃO  
MESTRADO PROFISSIONAL EM MATEMÁTICA  
EM REDE NACIONAL

LUIZ WEBERSON ALVES FELIPE

METAPOST NO ENSINO MÉDIO

Orientadora: Profa. Dra. Elisabete Sousa Freitas

Dissertação apresentada ao Programa de Pós-Graduação em Matemática em Rede Nacional do Instituto de Matemática da Universidade Federal de Mato Grosso do Sul-INMA/UFMS como parte dos requisitos para obtenção do título de Mestre.

Campo Grande - MS

2019

## **Metapost no Ensino Médio**

**Luiz Weberson Alves Felipe**

Dissertação apresentada ao Programa de Pós-Graduação em matemática em Rede Nacional do Instituto de Matemática da Universidade Federal de Mato Grosso do Sul-INMA/UFMS como parte dos requisitos para obtenção do título de Mestre.

Aprovado pela Banca Examinadora:

Profa. Dra. Elisabete Sousa Freitas - UFMS

Prof. Dr. Fabricio Sérgio de Paula - UEMS

Profa. Dra. Rúbia Mara de Oliveira Santos - UFMS

Campo Grande – MS, 25 de Novembro de 2019

*A Deus, pelo amor incondicional demonstrado por mim ao entregar seu Precioso Filho  
pra morrer na cruz em meu lugar, me concedendo a Graça de viver uma vida de vitórias  
na Tua Presença, dedico.*

*“Pois Deus enviou o seu Filho ao mundo, não para condenar o mundo, mas para que este fosse salvo por meio dele..”*

João 3:17

# Agradecimentos

Inicialmente, agradeço a Deus Pai pelo Seu infinito amor para comigo, demonstrado quando enviou o que tinha de mais precioso, Seu Filho Jesus Cristo, pra morrer em meu lugar, pelos meus pecados, possibilitando assim, que eu pudesse desfrutar da verdadeira vida aqui nesse mundo finito e, principalmente da vida eterna em comunhão com Ele.

À minha esposa Viviane do Nascimento Costa, mulher virtuosa, a qual Deus me concedeu a graça de viver junto, pelo auxílio e pelo companheirismo que demonstra em todo momento de sua vida e, principalmente por me incentivar no desenvolvimento desse trabalho.

Aos meu filhos, Luiz Matheus Diniz Felipe e Samuel Costa Felipe, aos quais dedico toda a vontade de viver, sendo assim, grandes motivações para não desistir.

Aos meus pais, Cirio Ferreira Felipe, Auride Alves e Edna dos Santos Oliveira, por tudo o que são na minha vida e por tudo o que passaram para possibilitar os cuidados necessários durante minha juventude.

À professora Elisabete Freitas pelo apoio, dedicação, confiança e orientação nesse trabalho.

Ao professor Marcelo Furquim, por contribuir com a cedência de suas aulas para trabalhar o METAPOST com os alunos.

Ao colega Vinícius Estevão, por colaborar grandemente no auxílio com o METAPOST.

Aos meus colegas de turma no PROFMAT, com os quais pude obter, compartilhar e, assim, aprender muitos conteúdos, necessários para o cumprimento desse trabalho, bem como no auxílio ao meu trabalho como docente.

Ao PROFMAT - Mestrado Profissional em Matemática em Rede Nacional pela oportunidade oferecida para o meu crescimento profissional.

# Resumo

O METAPOST é uma linguagem gráfica criada por John Douglas Hobby e proveniente de uma linguagem de programação chamada METAFONT, desenvolvida por Donald Knuth, matemático e cientista da computação, que possibilitou o desenvolvimento de ferramentas que permitem a criação de documentos científicos, padronizados e com excelente qualidade gráfica. É possível, com o uso METAPOST criar imagens de qualidade para textos matemáticos, usando uma técnica de programação que envolve cálculos matemáticos.

Este trabalho visa apresentar os conhecimentos básicos sobre coordenadas no plano e dos comandos utilizados no METAPOST, para o desenvolvimento de figuras geométricas e gráficos de funções reais presentes no Ensino Médio.

**Palavras-chave:** METAPOST; Figuras Geométricas; Funções Reais.

# Abstract

METAPOST is a graphic language created by John Douglas Hobby and coming from a programming language called METAFONT, developed by Donald Knuth, a mathematician and computer scientist, has enabled the development of tools that allow the creation of scientific documents, standardized and with excellent graphic quality. Using METAPOST you can create quality images for math texts using a programming technique that involves mathematical calculations.

This work aims to present the basic knowledge about coordinates in the plane and the commands used in METAPOST, for the development of geometric figures and graphs of real functions present in high school.

**Keywords:** METAPOST; Geometric Figures; Transformations; Real Functions.



# Sumário

<b>1</b>	<b>Coordenadas no Plano</b>	<b>3</b>
1.1	Coordenadas Cartesianas . . . . .	3
1.1.1	Coordenadas na Reta . . . . .	3
1.1.2	Coordenadas no Plano . . . . .	4
1.1.3	Distância entre dois pontos . . . . .	6
1.1.4	Ponto médio de um segmento . . . . .	8
1.2	Equações e gráficos . . . . .	9
1.2.1	Equação da reta . . . . .	9
1.2.2	Intersecção entre retas no plano . . . . .	10
1.3	Equação da circunferência . . . . .	11
<b>2</b>	<b>Comandos básicos do METAPOST</b>	<b>14</b>
2.1	Marcando pontos no plano . . . . .	14
2.2	Unidades de medida . . . . .	14
2.3	Largura de linhas . . . . .	15
2.4	Construção de segmentos . . . . .	16
2.5	Construção de polígonos . . . . .	17
2.6	Construção de circunferências . . . . .	17
2.7	Rótulos . . . . .	18
2.8	Setas . . . . .	19
2.9	Tipos de dados . . . . .	20
2.9.1	Dados tipo numeric . . . . .	21
2.9.2	Dados tipo pair . . . . .	21
2.9.3	Dados tipo path . . . . .	22
2.9.4	Dados tipo picture . . . . .	23

## SUMÁRIO

2.9.5	Dados tipo transform . . . . .	23
2.10	Utilização de cores em figuras . . . . .	23
2.11	Operadores matemáticos . . . . .	25
2.12	Laço for . . . . .	26
<b>3</b>	<b>Funções e Gráficos</b>	<b>28</b>
3.1	Funções . . . . .	28
3.1.1	Conjunto Imagem e Gráficos . . . . .	29
3.1.2	Parábolas . . . . .	31
3.1.3	Funções polinomiais . . . . .	36
3.2	Construção do Plano Cartesiano . . . . .	36
3.3	Construção de gráficos de funções reais básicas . . . . .	37
<b>4</b>	<b>Transformações com o METAPOST</b>	<b>43</b>
4.1	Transformação no Plano Cartesiano . . . . .	43
4.2	Homotetia . . . . .	44
4.3	Translação . . . . .	45
4.4	Rotação . . . . .	46
<b>5</b>	<b>Construindo figuras com o METAPOST</b>	<b>48</b>
5.1	Atividade 1: Construção de segmentos . . . . .	48
5.2	Atividade 2: Construção de eixos . . . . .	50
5.3	Atividade 3: Construção de ângulos . . . . .	51
5.4	Atividade 4: Construção de circunferências . . . . .	52
5.5	Atividade 5: Construção de triângulos . . . . .	54
5.6	Atividade 6: Construção de gráfico de função afim . . . . .	56
5.7	Atividade 7: Construção de gráficos de funções quadráticas usando trans- formações. . . . .	58
5.8	Relato da prática com os alunos . . . . .	62
<b>A</b>	<b>Como Trabalhar com o METAPOST</b>	<b>66</b>
A.1	Programas necessários . . . . .	66
A.2	Estrutura de um documento METAPOST . . . . .	67
A.3	Inserção de figuras num documento T <sub>E</sub> X . . . . .	68

## SUMÁRIO

A.4	Visualização de figuras METAPOST em plataforma on-line . . . . .	68
-----	--	----

# Lista de Figuras

1.1	Sistema de eixos ortogonais . . . . .	4
1.2	Representação do ponto A, de coordenadas (1, 3) . . . . .	5
1.3	Gráfico da reta vertical $x = 1$ . . . . .	5
1.4	Gráfico da reta horizontal $y = 3$ . . . . .	6
1.5	Gráfico da reta $y = 2x$ . . . . .	6
1.6	Distância entre os pontos $P_1 = (x_1, y_1)$ e $P_2 = (x_2, y_2)$ . . . . .	7
1.7	Distância entre os pontos C=(-2,3) e D=(2,1) do plano . . . . .	7
1.8	$M = (x_m, y_m)$ é o ponto médio do segmento $\overline{PQ}$ , pois $d(P, M) = d(Q, M)$ . . . . .	8
1.9	O ponto médio M=(1,-1) do segmento $\overline{AB}$ . . . . .	9
1.10	Representação da reta r que passa pelos pontos A=(2,-1) e B=(3,2) . . . . .	10
1.11	Intersecção I=(1,2) entre as retas r e s. . . . .	11
1.12	Representação da circunferência $\lambda$ de centro C=(1,2) e raio r igual a 2 . . . . .	13
2.1	Traçados de diferentes larguras. . . . .	16
2.2	Segmento de reta de extremidades (0,0) e (2,1) . . . . .	16
2.3	Tipos de traços . . . . .	17
2.4	Polígono fechado . . . . .	17
2.5	Circunferência de centro na origem e diâmetro u . . . . .	17
2.6	Sufixos usados no comando label . . . . .	18
2.7	Rótulos nos extremos do segmento $\overline{AB}$ de medida k . . . . .	19
2.8	Seta com extremidade em uma ponta . . . . .	19
2.9	Seta com extremidade nas duas pontas . . . . .	19
2.10	Sistema de eixos ortogonais formando o plano cartesiano . . . . .	20
2.11	Segmento $\overline{z_1 z_2}$ . . . . .	22

## LISTA DE FIGURAS

2.12	Curva ligando A e B, modificadas as direções de saída e de chegada e armazenadas numa variável m, tipo <code>path</code> .	22
2.13	Caminho fechado formando o polígono OABC	23
2.14	Caminho fechado usando <code>cycle</code> , formando o polígono OABC	24
2.15	Caminho fechado usando <code>cycle</code> , formando o polígono OABC	24
2.16	Caminho fechado usando <code>cycle</code> , formando o polígono OABC	24
2.17	Utilização de cores em linhas	25
2.18	Grade de pontos com 4 linhas e 4 colunas.	26
2.19	Plano cartesiano com grade tracejada.	27
2.20	Escada formada por segmentos horizontais.	27
3.1	Gráfico da função real $y = f(x) = mx + n$	30
3.2	Gráfico da função constante $y = n$	30
3.3	Gráficos das funções $y = \pm\sqrt{9 - x^2}$	31
3.4	Gráfico da função $y = f(x) = x^2$ .	31
3.5	Comparação dos gráficos das funções $y = x^2$ e $y = 2x^2$ .	32
3.6	Comparação dos gráficos das funções $y = x^2$ , $y = 2x^2$ e $y = \frac{1}{2}x^2$	32
3.7	Comparação dos gráficos das funções $y = -x^2$ , $y = -2x^2$ e $y = -\frac{1}{2}x^2$ .	33
3.8	Comparação dos gráficos das funções $y = x^2$ e $y = x^2 + 2$ .	33
3.9	Comparação dos gráficos das funções $y = x^2$ e $y = x^2 - 3$ .	34
3.10	Comparação dos gráficos das funções $y = x^2$ e $y = (x - 3)^2$ .	34
3.11	Gráficos de parábolas.	35
3.12	Gráfico da função polinomial $y = g(x) = x^3 - 3x - 2$	36
3.13	Plano cartesiano limitado pelo intervalo $[-4, 4]$	37
3.14	Gráfico da função real $y = f(x) = 2x$ no intervalo $[-1, 1]$ .	38
3.15	Gráfico da função real $y = f(x) = x^2 + 1$ no intervalo $[-2, 2]$ .	38
3.16	Gráfico da função polinomial $y = f(x) = -x^4 + 10x^3 - 35x^2 + 50x - 24$ no intervalo $]0, 5[$ .	39
3.17	Gráficos da função real $y = f(x) = 3x$ no intervalo $[-1, 1]$ , extrapolado (à esquerda) e cortado (à direita).	39
3.18	Intersecção I entre curvas no plano.	40
3.19	Intersecção Q entre curvas no plano.	41
3.20	Intersecção P entre retas no plano.	42

## LISTA DE FIGURAS

3.21	Intersecção I entre as curvas de equações $y = \sqrt{x}$ e $y = x^2$ . . . . .	42
4.1	Gráfico da função $f$ e da função $f^{-1}$ obtida de $f$ usando a transformação $T(x, y) = (y, x)$ . . . . .	44
4.2	Homotetia no triângulo $\triangle ABC$ . . . . .	45
4.3	Translação na circunferência $C_1$ . . . . .	46
4.4	Rotação de $60^\circ$ em torno da origem no triângulo $\triangle ABC$ . . . . .	46
4.5	Rotação de $60^\circ$ em torno do ponto $(2,2)$ . . . . .	47
4.6	Rotação de $120^\circ$ em torno do ponto $(1,2)$ . . . . .	47
5.1	Segmento $\overline{MN}$ de medida 5 cm . . . . .	49
5.2	Segmento tracejado $\overline{EF}$ de medida $\sqrt{13}$ cm . . . . .	49
5.3	Segmento pontilhado $\overline{CD}$ de medida $\sqrt{20}$ cm . . . . .	49
5.4	Eixo horizontal Ox . . . . .	50
5.5	Eixo vertical Oy . . . . .	50
5.6	Sistema de coordenadas cartesianas ortogonais Oxy . . . . .	51
5.7	Ângulo $\hat{A}OB$ de medida $\alpha = 30^\circ$ . . . . .	51
5.8	Ângulo $\hat{P}QR$ de medida $\theta = 75^\circ$ . . . . .	52
5.9	Ângulo $\hat{X}YZ$ de medida $\beta = 140^\circ$ . . . . .	52
5.10	Circunferência de centro na origem e raio 2cm . . . . .	53
5.11	Circunferência de centro $(2, 1)$ e raio 1,5 cm . . . . .	53
5.12	Circunferência de centro $(1, 1)$ e raio 1 preenchida na cor verde . . . . .	54
5.13	Triângulo $\triangle ABC$ escaleno. . . . .	55
5.14	Triângulo $\triangle PQR$ equilátero de lado 4 cm. . . . .	55
5.15	Triângulo $\triangle MNO$ isósceles de base 2 cm . . . . .	55
5.16	Gráfico da função real definida por $y=x-1$ na cor azul . . . . .	56
5.17	Gráfico da função real $y=-2x$ na cor vermelha . . . . .	57
5.18	Gráfico da função real $y = \frac{2}{3}x + 1$ na cor verde . . . . .	57
5.19	Gráfico da função $y = 2x^2$ na cor azul . . . . .	59
5.20	Gráfico da função $f(x) = -2(x + 1)^2$ . . . . .	60
5.21	Gráfico do resultado da rotação de $30^\circ$ no sentido anti-horário ao gráfico de $f(x) = (x - 1)^2 + 2$ . . . . .	61
5.22	Alunos trabalhando no laboratório. . . . .	64

## LISTA DE FIGURAS

5.23	Aluno desenvolvendo figura livremente. . . . .	64
A.1	Inserção de figura num documento $\text{T}_{\text{E}}\text{X}$ . . . . .	68
A.2	Visualizador $\text{METAPOST}$ on-line em branco . . . . .	69
A.3	Visualizador $\text{METAPOST}$ on-line com exemplo . . . . .	69

# Introdução

As tecnologias estão cada vez mais presente no nosso cotidiano, provocando grandes impactos em muitos aspectos do comportamento humano. Com o avanço da tecnologia surgem novas possibilidades no processo de ensino e aprendizagem, proporcionando aos docentes encontrar novas formas de ensinar e aos alunos novas formas de aprender.

Neste sentido, este trabalho apresenta uma maneira de utilização dos recursos tecnológicos para o aprendizado da matemática, particularmente no estudo de geometria analítica, ou seja, estudo da geometria no plano com a utilização de coordenadas.

O recurso utilizado é o METAPOST uma linguagem gráfica de programação, derivada do METAFONT e criada por John Douglas Hobby. Essa linguagem é utilizada para criar imagens em PostScript e é baseada em linhas de comando contendo cálculos matemáticos. As imagens geradas pelo METAPOST podem ser visualizadas em qualquer visualizador PostScript e em qualquer arquivo de saída obtido através de um arquivo TEX.

O objetivo principal é apresentar uma forma do aluno ter contato, aplicar e aprofundar alguns conceitos matemáticos estudados no Ensino Médio, estimulando o raciocínio lógico matemático e a organização de ideias para resolver problemas.

Sendo assim, o presente trabalho proporciona ao aluno do ensino médio, uma metodologia ativa no aprendizado sobre geometria, estudando figuras básicas através de coordenadas no plano e aprendendo os comandos utilizados no METAPOST, para a criação de figuras geométricas como polígonos, ângulos, circunferências e gráficos de funções reais.

Num primeiro momento, no Capítulo 1, apresentamos os conceitos básicos sobre Geometria Analítica, pois, o METAPOST se baseia na utilização de coordenadas no plano para o desenho de figuras.

No Capítulo 2, são descritos os comandos básicos usados pelo METAPOST na elaboração de figuras. Nos capítulos seguintes, 3 e 4, serão apresentados, respectivamente, exemplos de elaboração de gráficos de algumas funções estudadas no Ensino Médio e co-



mandos para realizar transformações nas figuras produzidas, como translação, rotação e homotetia.

O Capítulo 5 apresenta atividades para os alunos, bem como suas respectivas sugestões de soluções.

Este trabalho ainda apresenta um Apêndice ensinando como trabalhar com o METAPOST e inserir figuras num documento TEX.

As atividades propostas foram aplicadas a estudantes do 2º ano do Ensino Médio concomitante com o Curso Técnico em Programação de Jogos Digitais, da Escola Estadual Hércules Maymone em Campo Grande, escolhidos estrategicamente por terem familiaridade e pré-requisitos sobre programação de computadores.

# Capítulo 1

## Coordenadas no Plano

Neste capítulo, será apresentado um resumo básico de conceitos que serão utilizados na construção de figuras planas com o uso do METAPOST. As principais referências bibliográficas utilizadas neste capítulo são LIMA [9] e MUNIZ NETO [1].

### 1.1 Coordenadas Cartesianas

Indicamos por  $\mathbb{R}^2$  o conjunto formado pelos pares ordenados  $(x, y)$ , onde  $x$  e  $y$  são números reais. Assim

$$\mathbb{R}^2 = \{(x, y) \mid x, y \in \mathbb{R}\}$$

Dados  $(x_1, y_1)$  e  $(x_2, y_2)$  em  $\mathbb{R}^2$ , temos que  $(x_1, y_1) = (x_2, y_2)$  se, e somente se,  $x_1 = x_2$  e  $y_1 = y_2$ . Por exemplo,  $(3, 2) \neq (2, 3)$ .

#### 1.1.1 Coordenadas na Reta

Um eixo é uma reta na qual se fixou um ponto  $O$ , chamado a origem, e na qual se escolheu um sentido de percurso chamado positivo, sendo que o sentido inverso é dito negativo. Num eixo, dizemos que o ponto  $B$  está à direita do ponto  $A$ , ou equivalentemente, o ponto  $A$  está à esquerda do ponto  $B$ , quando o percurso de  $A$  para  $B$  na reta é positivo.

Todo eixo  $r$  pode ser posto, de modo natural, em correspondência com o conjunto  $\mathbb{R}$  dos números reais. À origem do eixo corresponde o número real zero. A cada ponto  $X$  de  $r$  à direita de  $O$  corresponde um número real positivo  $x$ , o qual é a distância  $d(O, X)$  de  $O$  à  $X$ . Aos pontos situados à esquerda de  $O$  correspondem os números negativos, cujos

## CAPÍTULO 1. COORDENADAS NO PLANO

valores absolutos são as distâncias desses pontos à origem.

Assim, a cada ponto  $X$  do eixo  $r$  corresponde o número real  $x$  tal que,  $x = d(O, X)$  se  $X$  está à direita da origem  $O$  e  $x = -d(O, X)$  se  $X$  está à esquerda de  $O$ . O número real  $x$  é denominado a coordenada do ponto  $X$  do eixo  $r$ .

### 1.1.2 Coordenadas no Plano

Um sistema de eixos ortogonais num plano  $\alpha$  é um par de eixos, representados por  $OX$  e  $OY$ , do plano  $\alpha$ , que são perpendiculares e têm a mesma origem  $O$ . Dizemos que o eixo  $OX$  é horizontal e o eixo  $OY$  é vertical.

Um plano  $\alpha$  com um sistema de eixos ortogonais está, de modo natural, em correspondência biunívoca com  $\mathbb{R}^2$ , o conjunto de pares ordenados de números reais. A cada ponto  $P$  do plano, baixamos por ele retas paralelas aos eixos  $OX$  e  $OY$ , conforme Figura 1.1. Essas retas cortam os eixos em pontos cujas coordenadas são  $x$  e  $y$  respectivamente. Ao ponto  $P \in \alpha$  corresponde o par ordenado  $(x, y) \in \mathbb{R}^2$ . Reciprocamente, a cada par ordenado  $(x, y) \in \mathbb{R}^2$  corresponde o ponto  $P \in \alpha$ , interseção da reta paralela a  $OY$  traçada pelo ponto de coordenada  $x$  do eixo  $OX$ , com a reta paralela a  $OX$  traçada a partir do ponto de  $OY$  de coordenada  $y$ .

Observamos que, dados dois eixos concorrentes quaisquer, o procedimento descrito acima estabelece uma correspondência biunívoca entre pontos do plano e pares ordenados de números reais. Neste trabalho utilizaremos sempre eixos ortogonais.

Os números  $x$  e  $y$  do par  $(x, y)$  associado ao ponto  $P$  do plano são chamados de coordenadas do ponto  $P$ , onde denominaremos  $x$  de abscissa e  $y$  de ordenada.

Se um ponto  $P$  do plano corresponde a  $(x, y) \in \mathbb{R}^2$ , escrevemos  $P = (x, y)$ .

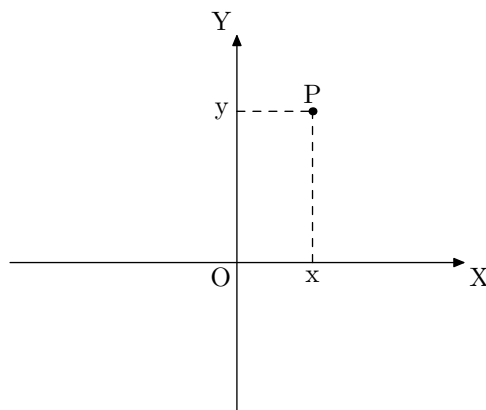


Figura 1.1: Sistema de eixos ortogonais

## CAPÍTULO 1. COORDENADAS NO PLANO

**Exemplo 1** O par ordenado  $A=(1,3)$  está associado ao ponto  $A$  do plano cartesiano, onde  $1$  é a abscissa e  $3$  é a ordenada, conforme Figura 1.2.

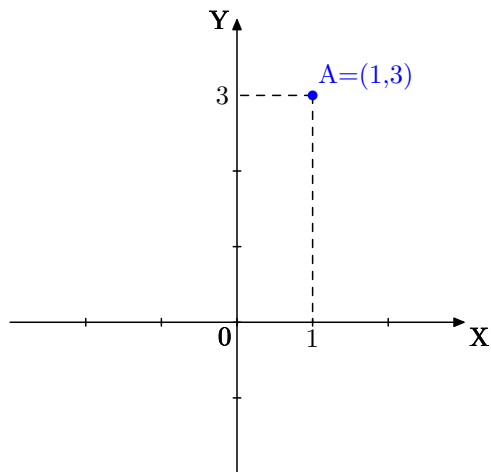


Figura 1.2: Representação do ponto  $A$ , de coordenadas  $(1, 3)$

**Exemplo 2** Representação dos pares ordenados  $(1, y)$ , onde  $y$  é um número real. Estes pares são descritos pela equação  $x = 1$ , conforme Figura 1.3.

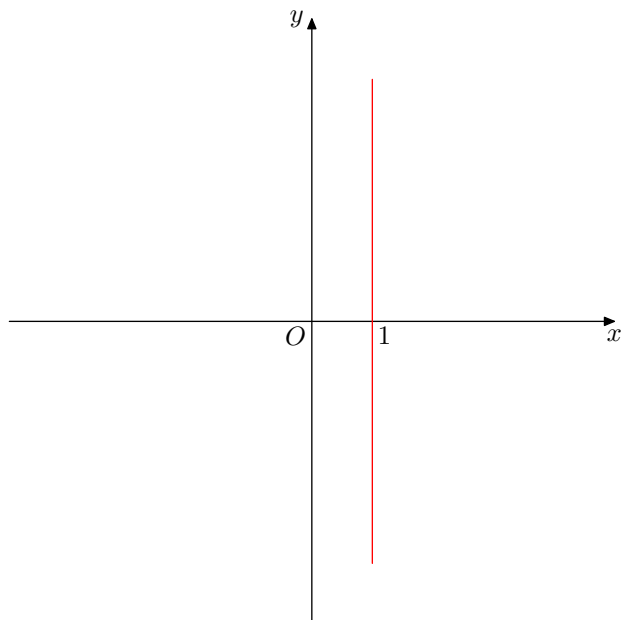


Figura 1.3: Gráfico da reta vertical  $x = 1$ .

**Exemplo 3** Representação dos pares ordenados  $(x, 3)$ , onde  $x$  é um número real. Estes pares são descritos pela equação  $y = 3$ , conforme Figura 1.4.

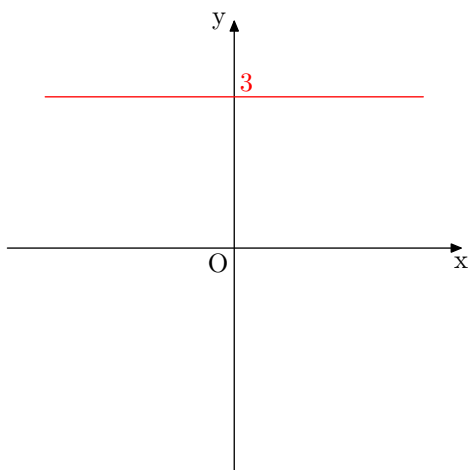


Figura 1.4: Gráfico da reta horizontal  $y = 3$ .

**Exemplo 4** *Representação dos pares ordenados  $(x, 2x)$ , onde  $x$  é um número real. Estes pares são descritos pela equação  $y = 2x$ , conforme Figura 1.5.*

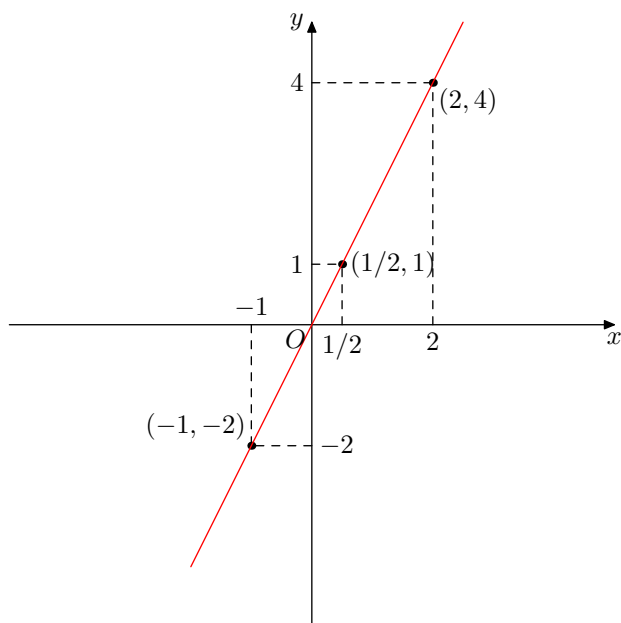


Figura 1.5: Gráfico da reta  $y = 2x$ .

### 1.1.3 Distância entre dois pontos

Considere dois pontos no plano  $P_1 = (x_1, y_1)$  e  $P_2 = (x_2, y_2)$ . A distância,  $d$ , entre  $P_1$  e  $P_2$  é mostrada na Figura 1.6 e dada pela seguinte fórmula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

CAPÍTULO 1. COORDENADAS NO PLANO

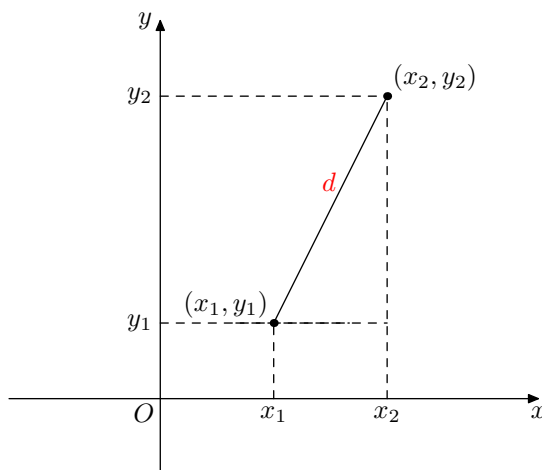


Figura 1.6: Distância entre os pontos  $P_1 = (x_1, y_1)$  e  $P_2 = (x_2, y_2)$ .

**Exemplo 5** A distância entre os pontos  $C$  e  $D$ , de coordenadas  $(-2, 3)$  e  $(2, 1)$  é igual a  $2\sqrt{5}$ , conforme calculado a seguir e mostrado na Figura 1.7.

$$d(C, D) = \sqrt{(x_d - x_c)^2 + (y_d - y_c)^2}$$

$$d(C, D) = \sqrt{(2 - (-2))^2 + (1 - 3)^2}$$

$$d(C, D) = \sqrt{16 + 4}$$

$$d(C, D) = \sqrt{20}$$

$$d(C, D) = 2\sqrt{5}$$

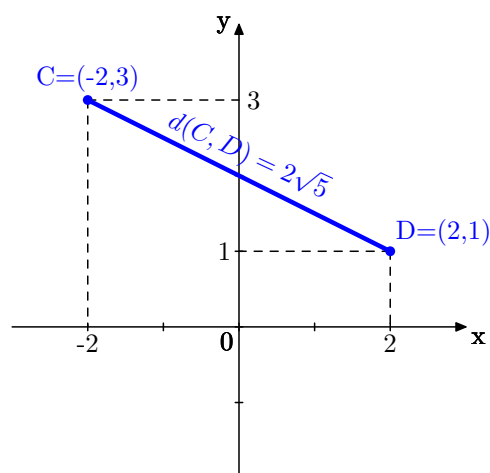


Figura 1.7: Distância entre os pontos  $C=(-2, 3)$  e  $D=(2, 1)$  do plano

### 1.1.4 Ponto médio de um segmento

Dado um segmento de reta  $\overline{PQ}$ , um ponto  $M$  pertencente a esse segmento é ponto médio de  $\overline{PQ}$  se a distância de  $P$  à  $M$  é igual à distância de  $Q$  à  $M$ , ou seja, se  $d(P, M) = d(Q, M)$ , conforme Figura 1.8. Temos que, se  $P = (x_p, y_p)$  e  $Q = (x_q, y_q)$  do plano  $\mathbb{R}^2$ , as coordenadas de  $M$ , ponto médio de  $\overline{PQ}$  são dadas por  $M = \left(\frac{x_p + x_q}{2}, \frac{y_p + y_q}{2}\right)$ . De fato, seja  $M = (x_m, y_m)$ , segue que

$$\begin{aligned} d(P, M)^2 &= (x_m - x_p)^2 + (y_m - y_p)^2 = \left(\frac{x_p + x_q}{2} - x_p\right)^2 + \left(\frac{y_p + y_q}{2} - y_p\right)^2 = \\ &= \left(\frac{x_q - x_p}{2}\right)^2 + \left(\frac{y_q - y_p}{2}\right)^2, \end{aligned}$$

$$\begin{aligned} d(Q, M)^2 &= (x_q - x_m)^2 + (y_q - y_m)^2 = \left(\frac{x_p + x_q}{2} - x_q\right)^2 + \left(\frac{y_p + y_q}{2} - y_q\right)^2 = \\ &= \left(\frac{x_p - x_q}{2}\right)^2 + \left(\frac{y_p - y_q}{2}\right)^2 \end{aligned}$$

e

$$d(P, M) + d(Q, M) = 2\sqrt{\frac{(x_p - x_q)^2 + (y_p - y_q)^2}{4}} = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2} = d(P, Q)$$

Portanto,  $d(P, M) = d(Q, M)$  e  $d(P, M) + d(Q, M) = d(P, Q)$ .

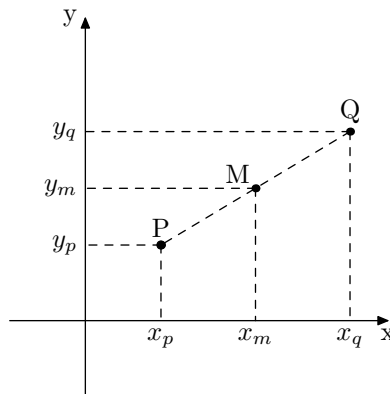


Figura 1.8:  $M = (x_m, y_m)$  é o ponto médio do segmento  $\overline{PQ}$ , pois  $d(P, M) = d(Q, M)$

**Exemplo 6** O ponto médio do segmento  $\overline{AB}$ , de coordenadas  $A=(-2,1)$  e  $B=(4,-3)$  é o ponto  $M=(1,-1)$ , conforme mostra a Figura 1.9.

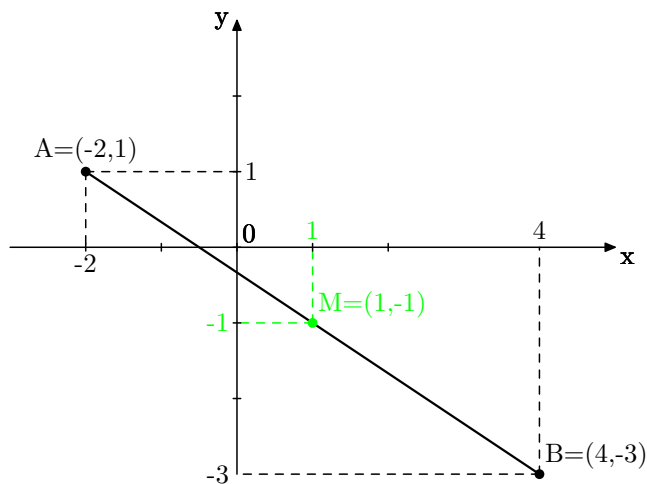


Figura 1.9: O ponto médio  $M=(1,-1)$  do segmento  $\overline{AB}$

## 1.2 Equações e gráficos

### 1.2.1 Equação da reta

Consideremos dois pontos distintos de uma reta  $r$ ,  $P = (x_1, y_1)$  e  $Q = (x_2, y_2)$ .

- (1) Se  $x_1 = x_2$ , a reta é vertical, descrita pela equação  $x = x_1$ .
- (2) Se  $x_1 \neq x_2$ , a reta é descrita pela equação  $y = mx + n$ , onde

$$m = \frac{y_2 - y_1}{x_2 - x_1} \text{ e } n = y_1 - mx_1$$

O número  $m$  da equação  $y = mx + n$  é denominado declive, coeficiente angular ou inclinação da reta.

De um modo geral, a equação de uma reta é dada por  $ax + by + c = 0$  onde  $a, b$  e  $c$  são constantes reais.

**Exemplo 7** Considere a equação  $ax + by + c = 0$ ,

com  $a = 0, b = 1$  e  $c = -1$ :

$$0 + y - 1 = 0 \iff y = 1 \text{ (reta horizontal)}$$



## CAPÍTULO 1. COORDENADAS NO PLANO

**Exemplo 8** Considere a equação  $ax + by + c = 0$ ,

com  $a = 1, b = 0$  e  $c = -1$ :

$$x - 1 = 0 \iff x = 1 \text{ (reta vertical)}$$

**Exemplo 9** Considere a equação  $ax + by + c = 0$ ,

com  $a = -2, b = 1$  e  $c = -1$ :

$$-2x + y - 1 = 0 \iff y = 2x + 1 \text{ (inclinação } m = 2)$$

**Exemplo 10** Na Figura 1.10 é representada a reta  $r$ , que passa pelos pontos  $A=(-2,-1)$  e  $B=(2,3)$ , cuja equação é dada por  $x - y + 1 = 0$ .

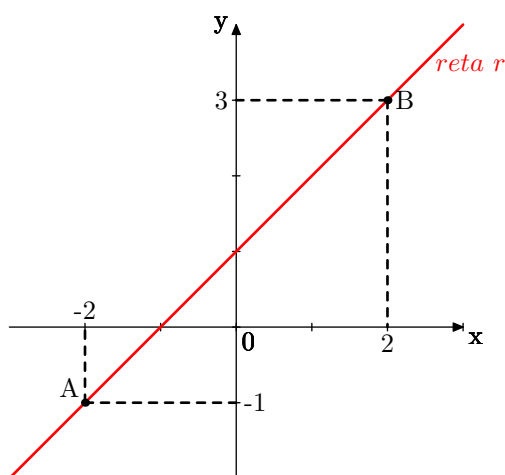


Figura 1.10: Representação da reta  $r$  que passa pelos pontos  $A=(2,-1)$  e  $B=(3,2)$

### 1.2.2 Intersecção entre retas no plano

Dadas duas retas concorrentes do plano  $\mathbb{R}^2$ , o ponto de encontro entre essas retas é chamado ponto de intersecção entre elas.

Deste modo, dadas as equações de duas retas concorrentes, não verticais, no plano  $\mathbb{R}^2$ , é possível obter as coordenadas do ponto de intersecção entre elas. Assim, sendo  $r : y = a.x + b$  e  $s : y = m.x + n$  as equações reduzidas das retas concorrentes  $r$  e  $s$ , pode-se obter a abscissa e a ordenada do ponto de intersecção  $I$  entre elas, resolvendo o sistema linear

$$\begin{cases} y = a.x + b \\ y = m.x + n \end{cases}$$

**Exemplo 11** Na Figura 1.11 são representadas as retas  $r$  e  $s$ , cujas equações reduzidas são dadas por  $r : y = x + 1$  e  $s : y = -x + 3$  e que se encontram no ponto  $I=(1,2)$ . De fato, resolvendo o sistema,

$$\begin{cases} y = x + 1 \\ y = -x + 3 \end{cases},$$

temos:

$$x + 1 = -x + 3$$

$$2x = 2$$

$$x = 1$$

Substituindo  $x = 1$  em qualquer das duas equações segue que  $I = (1, 2)$ .

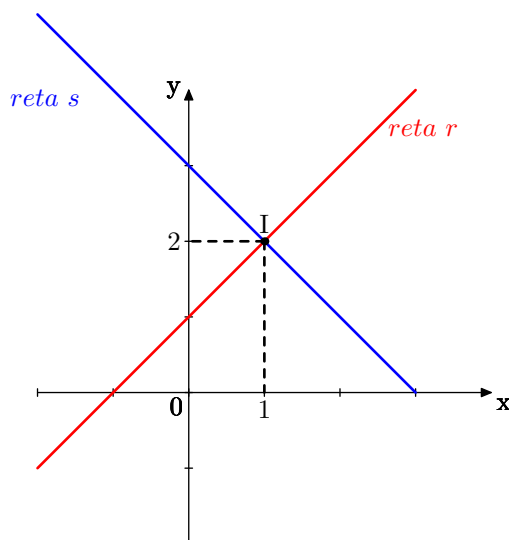


Figura 1.11: Intersecção  $I=(1,2)$  entre as retas  $r$  e  $s$ .

O METAPOST é capaz de encontrar pontos de intersecção entre duas curvas e marcá-lo, conforme código exposto na Figura 3.18.

### 1.3 Equação da circunferência

Uma circunferência qualquer é caracterizada por seu raio e seu centro. A circunferência de raio  $r$  e centro no ponto  $(a, b)$  é o conjunto de pontos  $(x, y)$ , cuja distância a  $(a, b)$  é igual a  $r$ . Usando a fórmula da distância temos,

$$(x - a)^2 + (y - b)^2 = r^2 \iff x^2 + y^2 - 2ax - 2by + a^2 + b^2 - r^2 = 0$$

## CAPÍTULO 1. COORDENADAS NO PLANO

**Exemplo 12** *Equação da circunferência de centro  $(2, -3)$  e raio 4:*

$$(x - 2)^2 + (x + 3)^2 = 16 \iff x^2 + y^2 - 4x + 6y - 3 = 0$$

**Exemplo 13** *Verifique se a equação  $x^2 + y^2 - 6x - 8y + 24 = 0$  representa uma circunferência. Em caso afirmativo, determine seu raio e centro.*

**solução:**

*Comparando a equação dada, com a equação geral de uma circunferência:*

$$x^2 + y^2 - 2ax - 2by + a^2 + b^2 - r^2 = 0$$

$$x^2 + y^2 \underbrace{-6}_{-2a}x \underbrace{-8}_{-2b}y + \underbrace{24}_{a^2+b^2-r^2} = 0$$

Daí,

$$\begin{cases} -2a = -6, \\ -2b = -8, \\ a^2 + b^2 - r^2 = 24 \end{cases}$$

Obtemos assim,  $a = 3, b = 4$  e  $9 + 16 - r^2 = 24$ , donde  $r^2 = 1$ . Portanto, temos uma circunferência de raio 1 e centro  $(3, 4)$ .

**Exemplo 14** *A equação  $x^2 + y^2 - 4x - 2y + 6 = 0$  não representa uma circunferência.*

*De fato, comparando com a equação geral:*

$$\begin{cases} -2a = -4, \\ -2b = -2, \\ a^2 + b^2 - r^2 = 6 \end{cases}$$

Daí,  $a = 2, b = 1$  e  $4 + 1 - r^2 = 6$ , donde  $r^2 = -1$ , o que não é possível. Observe que  $x^2 + y^2 - 4x - 2y + 6 = 0 \iff (x - 2)^2 + (y - 1)^2 = -1$ . Portanto a equação  $x^2 + y^2 - 4x - 2y + 6 = 0$  não tem solução.

## CAPÍTULO 1. COORDENADAS NO PLANO

**Exemplo 15** Na Figura 1.12 é representada a circunferência  $\lambda$  de centro  $C=(1,2)$  e raio  $r$  igual a 2, cuja equação é dada por  $(x - 1)^2 + (y - 2)^2 = 4$ .

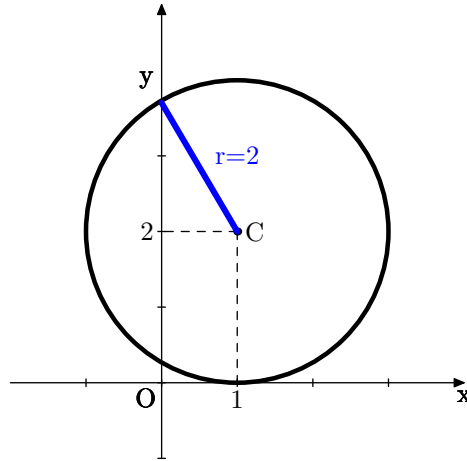


Figura 1.12: Representação da circunferência  $\lambda$  de centro  $C=(1,2)$  e raio  $r$  igual a 2

# Capítulo 2

## Comandos básicos do METAPOST

Neste capítulo, estudaremos os comandos básicos do METAPOST, a fim de produzir figuras básicas da geometria plana. Referências bibliográficas: VICENTE [8], CANDIA [7] e HOBBY [3].

### 2.1 Marcando pontos no plano

Para fazer um desenho, o METAPOST utiliza comandos do tipo `draw`. Por exemplo, para desenhar o ponto (1,2) utilizamos o comando

```
drawdot (1,2);
```

### 2.2 Unidades de medida

Se nenhuma unidade de medida for indicada, o METAPOST interpreta como sendo um `bp`, que é a sigla de “bit points”, sendo equivalente a uma unidade padrão de pontos PostScripts, cuja medida equivale a  $\frac{1}{72}$  de uma polegada.

O METAPOST possui unidades clássicas de medidas, conforme Tabela 2.1, algumas bem conhecidas como `cm` para centímetros, `in` para polegadas e `mm` para milímetros.

Unidade de Medida	Sigla	Conversão
pontos de impressão	pt	0,035145 cm
big points	bp	0,035278 cm
polegadas	in	2,54 cm
paicas	pc	0,423333 cm
milímetros	mm	0,1 cm
cíceros	cc	0,451167 cm
pontos Didot	dd	0,0376 cm

Tabela 2.1: Tabela de conversão das unidades de medida utilizadas no METAPOST

Pode-se estabelecer uma unidade, indicada por `u`, para servir de padrão em nosso desenho. Dessa forma é possível atribuir um valor para `u` e definir as coordenadas dos pontos em função de `u`. Por exemplo, para estabelecer uma unidade `u` com 2 cm, utiliza-se antes o comando

```
u=2cm;
```

e depois

```
draw (2u,u);
```

gerando um ponto de coordenadas (4cm, 2cm).

## 2.3 Largura de linhas

No METAPOST é possível mudar a largura de linhas utilizadas para desenhar segmentos, pontos ou fontes. Diversos comandos são utilizados para esse fim. Um deles é o `pencircle` que se refere ao formato da “caneta” utilizada para traçar linhas no desenho que, juntamente com o comando `scaled`, que diz a largura do traçado. Sendo assim, o comando

```
pickup pencircle scaled 0.5mm;
```

indica que, a partir dele, a ponta da caneta terá a forma circular e a largura das linhas desenhadas será de 0.5mm. Assim, a Figura 2.1 mostra diferentes larguras de traçados.

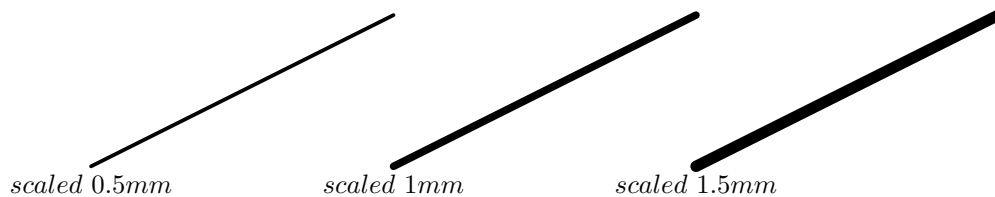


Figura 2.1: Traçados de diferentes larguras.

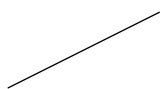
O comando `scaled` é usado para definir o tamanho do traçado e pode ser aplicado juntamente com vários comandos como o `draw`, o `label` entre outros.

## 2.4 Construção de segmentos

Para desenhar segmentos de reta, deve-se utilizar o comando `draw` seguido das coordenadas das extremidades do segmento separadas por `--`. Para desenhar o segmento da Figura 2.2, é utilizado o seguinte código:

```
draw (0,0)--(2cm,1cm);
```

Quando o sistema de eixos ortogonais não é estabelecido previamente, o METAPOST usualmente define um sistema de eixos ortogonais padrão em que a origem se encontra no canto inferior esquerdo da figura.

Figura 2.2: Segmento de reta de extremidades  $(0,0)$  e  $(2,1)$ 

É possível obter linhas tracejadas ou pontilhadas, utilizando o comando `dashed` da seguinte maneira

```
draw <figura> dashed <tipo de linha> scaled <tamanho>;
```

onde `<figura>` se refere ao desenho a ser feito e `<tipo de linha>` ao tipo de traço utilizado, podendo ser `evenly` para tracejado e `withdots` para pontilhados.

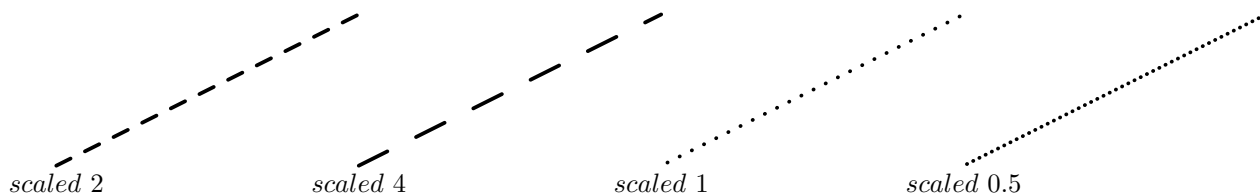


Figura 2.3: Tipos de traços

onde o termo `scaled` dentro do comando `dashed` define o espaçamento entre os traços ou pontos das linhas, conforme Figura 2.3.

## 2.5 Construção de polígonos

Para desenhar polígonos, basta lembrar que os mesmos são formados por segmentos de reta ligados pela origem de um e a extremidade de outro, deve-se utilizar o comando `draw` seguido das coordenadas das extremidades do segmento separadas por `--`. Para desenhar o polígono da Figura 2.4, utiliza-se o seguinte comando:

```
draw (0,0)--(3cm,0)--(2cm, 2cm)--(0,3cm);
```

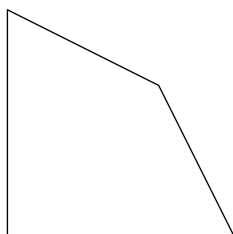
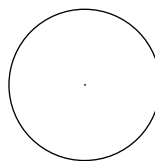


Figura 2.4: Polígono fechado

## 2.6 Construção de circunferências

Para desenhar uma circunferência de centro na origem e diâmetro `u` usamos o comando `draw fullcircle scaled u;`, conforme Figura 2.5.

```
u:=1cm;
draw fullcircle scaled u;
drawdot origin;
```

Figura 2.5: Circunferência de centro na origem e diâmetro `u`



## 2.7 Rótulos

Para integrar textos com figuras, permitindo indicar nomes para as mesmas é possível utilizar rótulos, com os comandos `label` e `dotlabel`. O comando `label` é utilizado para rotular figuras no desenho, por outro lado, o comando `dotlabel` é específico para marcar pontos dados por suas coordenadas.

A utilização desse comando é feita da seguinte maneira:

```
label.<sufixo> (<nome do rótulo>, <posição>);
```

O termo `<sufixo>` se refere à posição em que o rótulo será colocado na figura. Estes sufixos são dispostos conforme Figura 2.6.

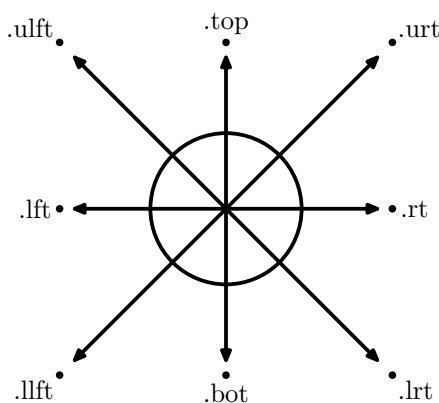


Figura 2.6: Sufixos usados no comando `label`

Se a posição do rótulo não for especificada, o mesmo será centralizado

Já o termo `<nome do rótulo>` representa ao texto que deverá ser escrito no rótulo. Se posto entre aspas duplas, o texto mostrado no rótulo será “idêntico” ao texto escrito. Por outro lado, podemos utilizar o comando `btex <texto do rótulo> etex`, onde o termo `<texto do rótulo>` poderá ser uma instrução escrita no ambiente matemático do  $\text{T}_{\text{E}}\text{X}$ , possibilitando assim, escrever expressões matemáticas nos rótulos.

Por exemplo, para dar nomes aos pontos que são os extremos de um segmento, cujas coordenadas são  $(0, 0)$  e  $(2, 1)$ , utilizamos os comandos

Note que o rótulo `k` na Figura 2.7, que dá o nome do segmento  $\overline{AB}$  foi posicionado no ponto médio de  $A$  e  $B$ .

```

u=2cm;
pickup pencircle scaled 0.5mm;
draw (0,0)--(2u,u);
label.lft("A", origin);
label.rt("B", (2u,u));
label.ulft("k", (u,u/2));

```

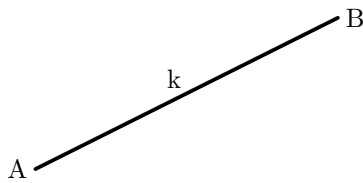


Figura 2.7: Rótulos nos extremos do segmento  $\overline{AB}$  de medida  $k$

## 2.8 Setas

É muito importante o uso de setas para o esboço de alguns desenhos como eixos em gráficos e vetores. No METAPOST, podemos usar os comandos `drawarrow` e `drawdblarrow` para esse fim, para caminhos retilíneos e curvos. O comando `drawarrow` coloca uma seta numa extremidade do segmento especificado, enquanto `drawdblarrow` coloca setas nas duas extremidades. Por exemplo, o comando

```
drawarrow origin--(3cm, 2cm);
```

desenha uma seta com origem no ponto  $(0, 0)$  e extremidade no ponto  $(3, 2)$ , como indica a Figura 2.8.

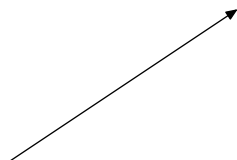


Figura 2.8: Seta com extremidade em uma ponta

O comando `drawdblarrow` coloca setas nas duas extremidades, como indica a Figura 2.9.

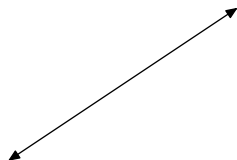


Figura 2.9: Seta com extremidade nas duas pontas

Dessa forma, podemos criar os eixos coordenados, formando o plano cartesiano, como na Figura 2.10.

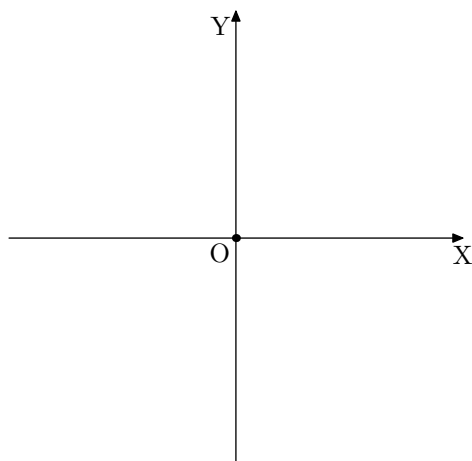


Figura 2.10: Sistema de eixos ortogonais formando o plano cartesiano

## 2.9 Tipos de dados

Os tipos de dados são usados para armazenar informações como medidas, partes de figuras, partes de códigos, entre outros, a fim de utilizá-las em algum momento da programação. No METAPOST existem dez tipos de dados usados para armazenar informações, como mostra a Tabela 2.2.

Tipo de dado	Utilização
<code>numeric</code>	armazenar variáveis numéricas
<code>pair</code>	armazenar pares ordenados
<code>path</code>	armazenar caminhos entre pontos
<code>transform</code>	armazenar transformações
<code>rgbcolor</code>	armazenar cores rgb
<code>cmykcolor</code>	armazenar cores cmyk
<code>string</code>	armazenar uma cadeia de caracteres
<code>boolean</code>	armazenar números booleanos
<code>picture</code>	armazenar figuras
<code>pen</code>	armazenar tipo de caneta

Tabela 2.2: Tipos de dados utilizados no METAPOST

Para representar o nome de uma variável, utilizamos qualquer combinação de caracteres simples, podendo ser minúsculos ou maiúsculos. Para declarar o tipo de dado de uma variável, basta digitar o tipo de dado seguido do nome da variável e um sinal “;”.

## CAPÍTULO 2. COMANDOS BÁSICOS DO METAPOST

Podemos declarar uma família de variáveis, utilizando colchetes após o nome da variável, como veremos mais adiante.

A seguir serão apresentados alguns tipos de dados (os principais a serem utilizados neste trabalho), como declará-lo e como utilizá-lo.

### 2.9.1 Dados tipo `numeric`

O tipo `numeric` é utilizado para armazenar variáveis numéricas. Com o comando

```
numeric var;
```

declaramos uma variável numérica cujo nome é `var`. Para atribuir um número (valor) à variável `var`, por exemplo o número 3 cm procedemos da seguinte maneira:

```
var:= 3cm;
```

### 2.9.2 Dados tipo `pair`

O tipo `pair` é utilizado para armazenar pares ordenados de números do tipo `numeric` numa variável. O código

```
pair A;
```

```
A:=(1cm,2cm);
```

associa o par `(1cm,2cm)` à variável `A`.

O METAPOST usa a mesma estrutura matemática  $(x, y)$  para definir esses pares, onde  $x$  é a abscissa e  $y$  é a ordenada de um ponto  $P$  do plano cartesiano. Assim, quando se define `P:=(2cm, 3cm);`, o METAPOST gera automaticamente duas variáveis numéricas `xpart P` e `ypart P`, respectivamente, valendo 2 cm e 3 cm. É intrínseco ao METAPOST que toda variável da forma `z<índice>` é do tipo `pair`, e isso só é válido utilizando a letra `z`. Por exemplo, a Figura 2.11 mostra o segmento  $\overline{z1z2}$ .

```
z1=(1cm,2cm);
z2=(4cm,1cm);
draw z1..z2;
dotlabel.llft("z1", (1cm,2cm));
dotlabel.lrt("z2", (4cm,1cm));
```

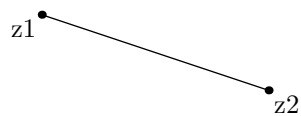


Figura 2.11: Segmento  $\overline{z_1z_2}$

Para usar outra letra com índices para representar pares, como, por exemplo, X1, X2, X3, e assim por diante, deve-se declarar essas variáveis com o comando `pair X[]`, que define qualquer variável do tipo `X[<índice>]` ou `X<índice>` como sendo um par ordenado.

### 2.9.3 Dados tipo path

O tipo `path` é utilizado para armazenar caminhos entre pontos, circunferências e outras figuras. Os caminhos entre pontos podem ser agrupados de duas maneiras, como o comando “-”, gerando um segmento de reta, e com o comando “..”, gerando uma curva entre um par de pontos.

Para usar o comando “..”, é possível determinar uma direção angular para um caminho num determinado ponto, por exemplo, de saída de um ponto e de chegada em outro ponto. Nesse procedimento, utiliza-se o comando `dir  $\theta$` , onde  $\theta$  é um ângulo medido em graus. Na Figura 2.12 temos um exemplo de como utilizar esse comando.

```
pair A, B; path m;
A:=origin; B:=(3cm, 4cm);
dotlabel.lrt("A", origin);
dotlabel.lft("B", (3cm,4cm));
m:=A{dir 135}..B{dir 180};
draw m;
```

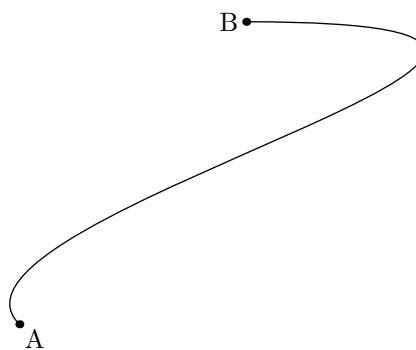


Figura 2.12: Curva ligando A e B, modificadas as direções de saída e de chegada e armazenadas numa variável m, tipo path.

Se declaramos uma variável do tipo `path` usando a notação, por exemplo, `q[ ]`, cria-se um grupo de variáveis, podendo armazenar caminhos nas variáveis indexadas `q[1]`, `q[2]`, `q[3]`, e assim por diante.

### 2.9.4 Dados tipo picture

O tipo `picture` é utilizado para armazenar figuras. Para esse fim, é necessário declarar uma variável `currentpicture`. Essa variável armazena a figura atual que foi desenhada com os comandos `fill`, `draw` e `filldraw`, seus respectivos comandos de desfazer as ações `unfill`, `undraw` e `unfilldraw`. Esse comando é mais útil para desenhos mais complexos.

Os comandos `fill` e `filldraw` serão vistos mais adiante na Seção 2.10.

### 2.9.5 Dados tipo transform

O tipo `transform` é utilizado para realizamos alterações em figuras como translações, rotações e dimensionamento. Esse comando guarda uma transformação em uma única variável a fim de ser utilizada posteriormente. O comando `transform` pode ser aplicado a pontos e também a caminhos definidos usando o comando `path`. Mais detalhes sobre como utilizar o comando `transform` será apresenta no Capítulo 4.

Os tipos da dados `pen`, `boolean` e `string` não serão utilizados no desenvolvimento deste trabalho, desta forma não serão estudados aqui.

## 2.10 Utilização de cores em figuras

Pode-se preencher, com cores, regiões fechadas, de figuras definidas no METAPOST. O comando utilizado para essa finalidade é o `fill` e, é aplicado para caminhos e curvas fechadas cujo código termina com o comando `cycle`. A seguir, apresentamos exemplos de como desenhar caminhos fechados com o METAPOST.

**Exemplo 16** *Sem utilização do comando `cycle` no final, conforme Figura 2.13.*

```
u:= 1 cm;
draw origin--(3u, 0)--(4u, 2u)--(u,u)--origin;
dotlabel.llft("O", origin);
label.bot("A", (3u, 0));
label.rt("B", (4u, 2u));
label.ulft("C", (u, u));
```

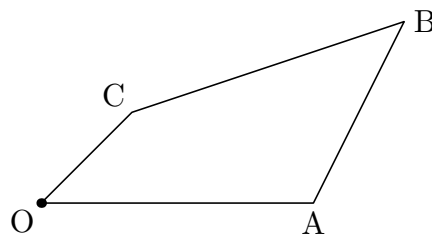


Figura 2.13: Caminho fechado formando o polígono OABC

## CAPÍTULO 2. COMANDOS BÁSICOS DO METAPOST

**Exemplo 17** Com utilização do comando `cycle` no final, conforme Figura 2.14.

```
u:= 1 cm;
draw origin--(3u, 0)--(4u, 2u)--(u,u)--cycle;
dotlabel.llft("O", origin);
label.bot("A", (3u, 0));
label.rt("B", (4u, 2u));
label.ulft("C", (u, u));
```

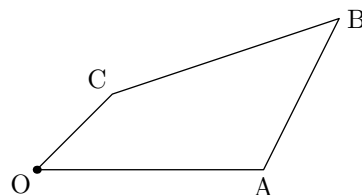


Figura 2.14: Caminho fechado usando `cycle`, formando o polígono OABC

**Exemplo 18** Com utilização do comando `cycle` no final.

```
u:= 1 cm;
draw origin--(3u, 0)--(4u, 2u)--(u,u)--cycle;
dotlabel.llft("O", origin);
label.bot("A", (3u, 0));
label.rt("B", (4u, 2u));
label.ulft("C", (u, u));
```

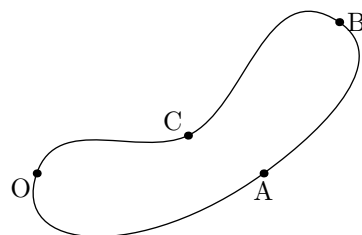


Figura 2.15: Caminho fechado usando `cycle`, formando o polígono OABC

O comando para preenchimento `fill` é usado pelo METAPOST em caminhos fechados, cujos comandos possuem `cycle` no final, como nos comandos mostrados nas Figuras 2.14 e 2.15. Para preencher o desenho da Figura 2.15 com a cor azul, procede-se conforme a figura 2.16.

```
u:= 1 cm;
fill origin..(3u, 0)..(4u, 2u)..(2u,0.5u)..cycle withcolor blue;
draw origin..(3u, 0)..(4u, 2u)..(2u,0.5u)..cycle;
dotlabel.llft("O", origin);
dotlabel.bot("A", (3u, 0));
dotlabel.rt("B", (4u, 2u));
dotlabel.ulft("C", (2u, 0.5u));
```

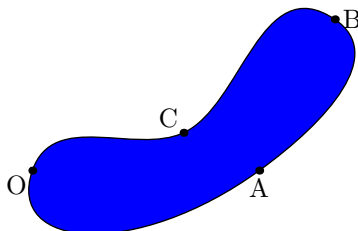


Figura 2.16: Caminho fechado usando `cycle`, formando o polígono OABC

O METAPOST trabalha de acordo com a ordem das linhas de comando. Assim, se o comando `fill` for colocado depois dos comandos `draw` e `dotlabel`, por exemplo, o preenchimento ficará sobreposto ao contorno da figura.

## CAPÍTULO 2. COMANDOS BÁSICOS DO METAPOST

No código da Figura 2.16 aparece o comando `withcolor` que serve para selecionar a cor utilizada no preenchimento da figura. Após o comando `withcolor` deverá aparecer o nome da cor escolhida, em inglês. O METAPOST possui as seguintes cores pré-definidas: `white`, para branco; `black`, para preto; `red`, para vermelho; `green`, para verde; `blue`, para azul; `yellow`, para amarelo; `magenta`, para magenta; e `cyan` para ciano.

Para colorir linhas, ao final do comando `draw`, acrescenta-se `withcolor <cor>`, conforme Figura 2.17.

Pode-se utilizar outras cores, além das pré-definidas pelo METAPOST, do sistema de cores RGB e CMYK. Tais cores não serão usadas neste trabalho, portanto não serão estudadas.

```
u:= 1cm;
drawarrow ((0, 5u)--(5u, 5u)) withcolor red;
draw (0,4u)--(5u,4u) dashed evenly withcolor green;
label.rt("label", (0, 3u)) withcolor blue;
draw (0,2u)..(u,u)..{dir -60}(4u, 2u) withcolor 0.7white;
```

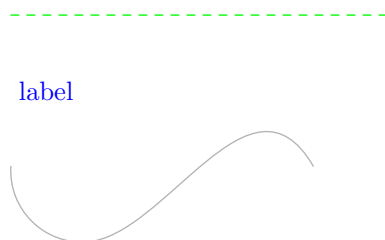


Figura 2.17: Utilização de cores em linhas

### 2.11 Operadores matemáticos

No METAPOST, existem alguns operadores matemáticos que são muito úteis no desenvolvimento de figuras. Para as operações de adição, subtração, multiplicação e divisão, são utilizados, respectivamente, `+`, `-`, `*` e `/`. Em potências, utiliza-se `3**2` para representar a potência  $3^2$ .

Lembrando que o METAPOST entende as operações na ordem em que aparecem os operadores, assim é necessário o uso dos parênteses. Ao digitar a sequência `3*x**2`, o METAPOST entende a expressão  $(3x)^2$ , e não  $3x^2$ . Se quiser a segunda opção, se deve digitar `3*(x**2)`.

Outros operadores importantes são `abs x` para representar  $|x|$  (módulo de  $x$ ), `sqrt x`



que representa  $\sqrt{x}$  (raiz quadrada de  $x$ ) e `a mod b` que significa o resto da divisão euclidiana de  $a$  por  $b$ .

Utilizando os comandos `sind x` e `cosd x`, são retornados, respectivamente os valores de seno e cosseno de um ângulo  $x$  em graus, importantes para cálculos trigonométricos.

Existem muitos outros operadores interpretados pelo METAPOST que não estão citados, como os operadores lógicos, entre outros. Mais detalhes sobre esses operadores pode ser encontrado em Hobby [13].

## 2.12 Laço for

Em toda linguagem de programação existem os “laços” (loopings em inglês) que são estruturas de repetição. As estruturas de repetição executam a repetição de um conjunto de instruções enquanto uma determinada condição é verdadeira.

No METAPOST o laço mais utilizado é o `for`, cuja estrutura é dada conforme a seguir:

```
for <contador> = <valor inicial> step <intervalo> until <valor final> :
<texto do laço>
endfor
```

Assim, um loop (laço) começando para  $i = \text{<valor inicial> step <intervalo> until <valor final>}$  varre uma sequência de valores  $i$  iguais a:  $a, a + b, a + 2b, \dots$ , onde  $a = \text{<valor inicial>}$  e  $b = \text{<intervalo>}$ , parando quando  $i$  assumir  $\text{<valor final>}$ .

**Exemplo 19** *Para montar uma grade de pontos no plano com 4 linhas e 4 colunas, fazemos como na figura 2.18:*

	•	•	•	•
<code>u:=1cm;</code>				
<code>pickup pencircle scaled 3bp;</code>	•	•	•	•
<code>for i=0 step 1 until 3:</code>				
<code>for j=0 step 1 until 3:</code>				
<code>drawdot (i*u,j*u);</code>				
<code>endfor</code>	•	•	•	•
<code>endfor</code>				
	•	•	•	•

Figura 2.18: Grade de pontos com 4 linhas e 4 colunas.

## CAPÍTULO 2. COMANDOS BÁSICOS DO METAPOST

**Exemplo 20** Na figura 2.19, temos o plano cartesiano com grade tracejada.

```
u:= 1 cm;
for i=-3u step u until 3u:
draw (i,-3u)--(i,3u) dashed evenly;
draw (-3u,i)--(3u,i) dashed evenly;
drawarrow ((-3u, 0)--(3u, 0));
drawarrow ((0, -3u)--(0, 3u));
label.rt("x", (3u, 0));
label.top("y", (0, 3u));
label.llft("O", (0, 0));
endfor;
```

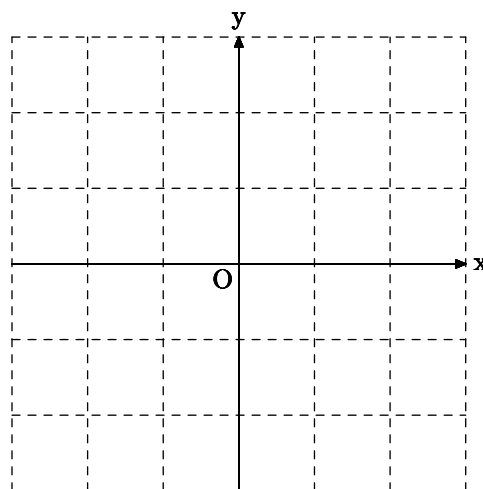


Figura 2.19: Plano cartesiano com grade tracejada.

**Exemplo 21** Neste caso, temos na figura 2.20 a seguir, um modelo de escada composta por segmentos horizontais.

```
u:=2cm;
pickup pencircle scaled 3bp;
for i=0 step 1 until 3:
for j=0 step 1 until 3:
drawdot (i*u,j*u);
endfor
endfor
```

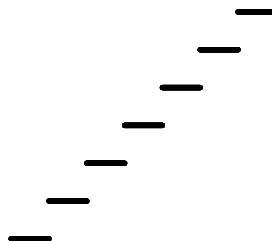


Figura 2.20: Escada formada por segmentos horizontais.

# Capítulo 3

## Funções e Gráficos

Neste capítulo, serão estudados os comandos do METAPOST que possibilitam construir gráficos de funções reais contínuas no plano cartesiano. Inicialmente, são abordadas noções sobre funções reais e, a seguir, são apresentados os principais comandos do METAPOST, utilizados na elaboração das funções afins e quadráticas. Referências bibliográficas: DELGADO [1], MUNIZ NETO [4] e HOBBY [13].

### 3.1 Funções

**Definição 1** *Dados os conjuntos  $A$ ,  $B$ , uma função  $f : A \rightarrow B$  é uma regra que diz como associar a cada elemento  $x \in A$  um elemento  $y \in B$ . O conjunto  $A$  chama-se o domínio e  $B$  o contra-domínio da função  $f$ . Para cada  $x \in A$ , o elemento  $y \in B$ , associado a  $x$ , é denotado por  $y = f(x)$  e é denominado a imagem de  $x$  pela função  $f$ . Dizemos que  $x$  é uma variável independente e  $y = f(x)$  uma variável dependente. A notação  $x \rightarrow f(x)$  indica que,  $f$  transforma (ou leva)  $x$  em  $f(x)$ .*

**Observação:** Em muitos exemplos de funções  $f : A \rightarrow B$ ,  $A$  e  $B$  são conjuntos numéricos e a regra  $x \rightarrow f(x)$  é dada por uma fórmula. Em geral, não precisa ser assim. A regra que estabelece como obter  $f(x)$  quando é dado  $x$  é inteiramente arbitrária, sendo sujeita a duas condições:

- a regra deve dizer quem é  $f(x)$ , para todo  $x \in A$  dado.
- não pode haver ambiguidades: a cada  $x \in A$ , a regra deve fazer corresponder um único  $f(x)$  em  $B$ .

## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

**Exemplo 22** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = 2x + 1$ . Neste caso, o domínio é o conjunto dos números reais e para cada  $x \in \mathbb{R}$  o valor de  $f(x) \in \mathbb{R}$  está bem definido. Por exemplo,  $f(20) = 41$ .

**Exemplo 23** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por

$$f(x) = \begin{cases} 2x, & \text{se } x \geq 0; \\ 3x + 1, & \text{se } x < 0. \end{cases}$$

Neste caso, a função foi definida usando duas expressões. Para calcularmos, por exemplo,  $f(-2)$ , observamos que  $-2 < 0$  e daí,  $f(-2) = 3 \cdot (-2) + 1 = -5$ .

**Exemplo 24** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = |x - 1|$ . Neste caso, podemos escrever

$$f(x) = \begin{cases} x - 1, & \text{se } x \geq 1; \\ 1 - x, & \text{se } x < 1. \end{cases}$$

### 3.1.1 Conjunto Imagem e Gráficos

Podemos visualizar uma função  $f : D \rightarrow \mathbb{R}$  como o conjunto de pares ordenados  $(x, f(x))$ , onde  $x$  varia no domínio de  $f$ , assim é correto escrever

$$f = \{(x, f(x)) : x \in D\}$$

Esse conjunto é também chamado de gráfico da função, tendo em vista que a representação desse conjunto no plano cartesiano nos dá o gráfico da equação  $y = f(x)$ . O conjunto imagem de uma função  $f$  é o conjunto de todos os valores  $f(x)$  para todos os possíveis valores de  $x$  em seu domínio. Costuma-se indicar por  $f(D)$  o conjunto imagem da função  $f$ :

$$f(D) = \{f(x) : x \in D\}$$

**Exemplo 25** As funções mais simples, que trataremos aqui, são aquelas dadas por uma equação do tipo

$$y = mx + n$$

onde  $m$  e  $n$  são constantes reais. Uma função desse tipo é chamada função afim. Como já vimos, o gráfico é uma reta de inclinação  $m$ , que corta o eixo das ordenadas no ponto

## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

de ordenada  $n$ . Veja Figura 3.1.

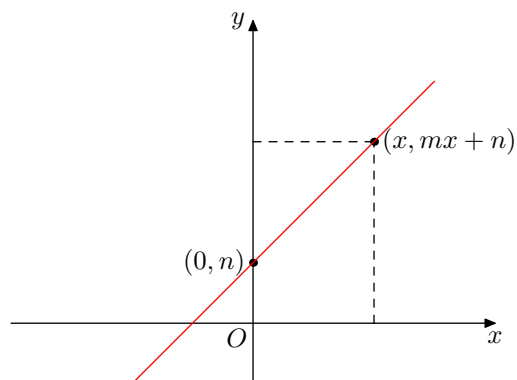


Figura 3.1: Gráfico da função real  $y = f(x) = mx + n$

Em particular, quando  $m = 0$ , temos a função constante  $y = n$ , cujo gráfico é uma reta horizontal, como mostra a Figura 3.2.

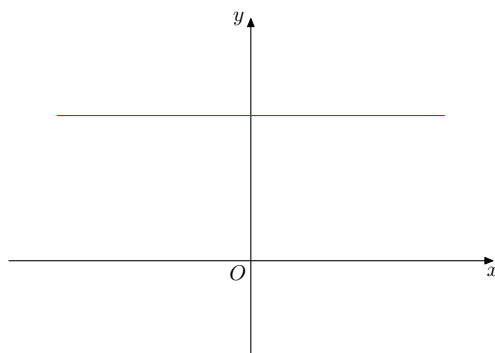


Figura 3.2: Gráfico da função constante  $y = n$

**Exemplo 26** A função valor absoluto é definida por  $f(x) = |x|$  para todo  $x \in \mathbb{R}$ , isto é,

$$f(x) = \begin{cases} x, & \text{se } x \geq 0; \\ -x, & \text{se } x < 0. \end{cases}$$

**Exemplo 27** A equação  $x^2 + y^2 = 9$  representa um círculo de centro na origem e raio 3. Resolvida em relação a  $y$ , ela é equivalente a

$$y = \pm\sqrt{9 - x^2}$$

o que nos permite definir duas funções:

$$f(x) = \sqrt{9 - x^2} \text{ e } g(x) = -\sqrt{9 - x^2}$$

## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

O domínio dessas duas funções é o intervalo  $[-3, 3]$  e seus gráficos são mostrados na Figura 3.3.

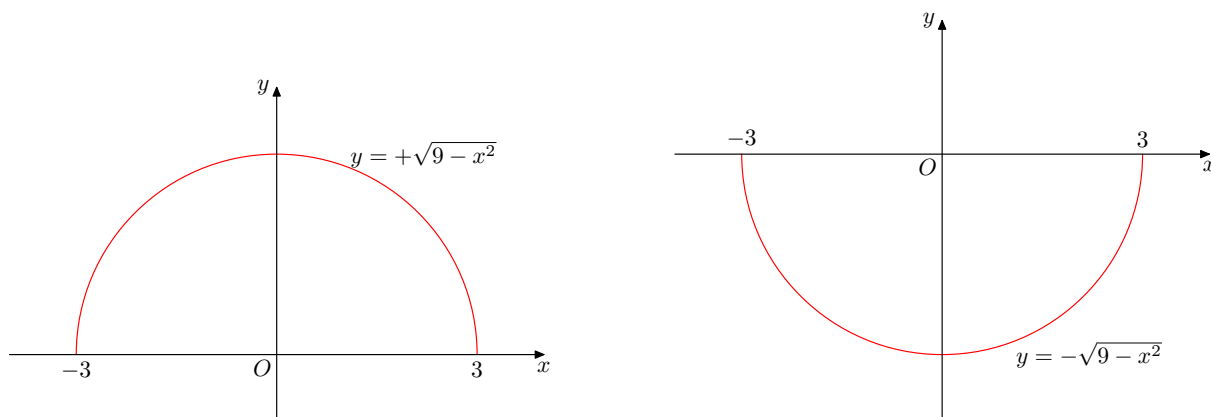


Figura 3.3: Gráficos das funções  $y = \pm\sqrt{9-x^2}$

### 3.1.2 Parábolas

Vamos estudar funções do tipo  $y = f(x) = ax^2 + bx + c$ , onde  $a \neq 0$ , começando pelo caso mais simples  $f(x) = x^2$ . Os gráficos dessas funções são parábolas.

**Exemplo 28** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = x^2$ . Calculando  $f(x)$  para alguns valores de  $x$ :

$$f(0) = 0, f\left(\frac{1}{2}\right) = f\left(-\frac{1}{2}\right) = \frac{1}{4}, f(1) = f(-1) = 1, f(2) = f(-2) = 4$$

Marcando esses pontos no plano cartesiano, já teremos uma ideia do gráfico da função, conforme Figura 3.4:

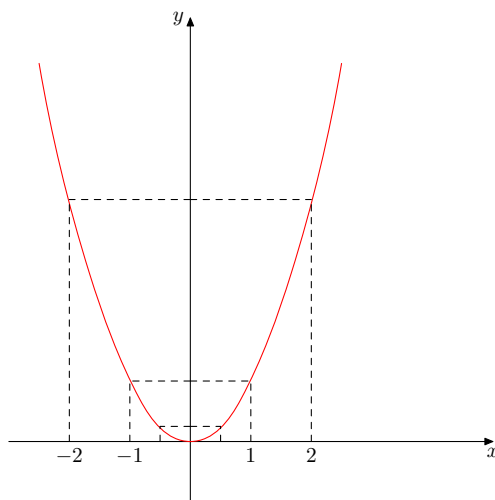


Figura 3.4: Gráfico da função  $y = f(x) = x^2$ .

### CAPÍTULO 3. FUNÇÕES E GRÁFICOS

**Exemplo 29** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = 2x^2$ . A Figura 3.5, nos mostra como é o gráfico da função  $f$  comparado com o gráfico da função  $y = x^2$ .

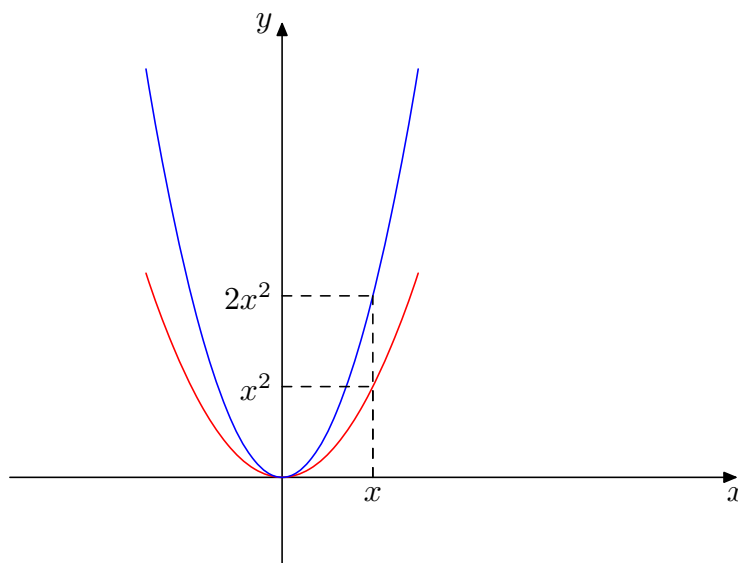


Figura 3.5: Comparação dos gráficos das funções  $y = x^2$  e  $y = 2x^2$ .

**Exemplo 30** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = \frac{1}{2}x^2$ . A figura 3.6, nos mostra como é o gráfico da função  $f$  comparado com os gráficos das funções  $y = x^2$  e  $y = 2x^2$ .

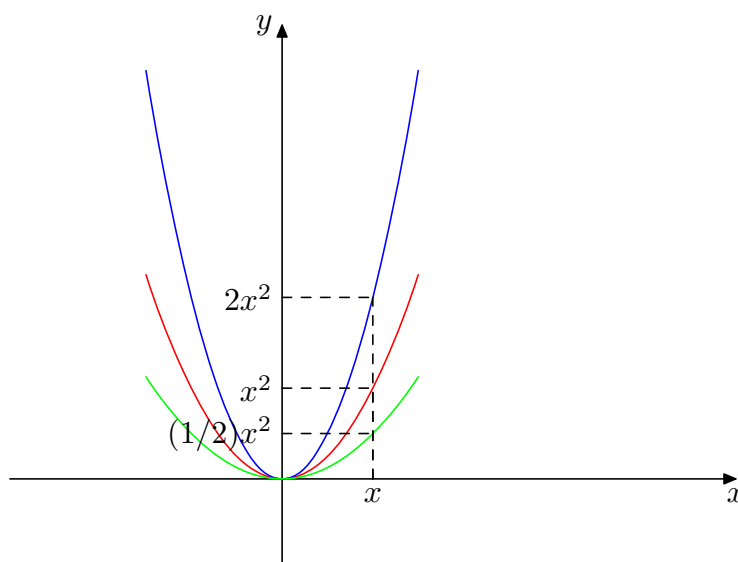


Figura 3.6: Comparação dos gráficos das funções  $y = x^2$ ,  $y = 2x^2$  e  $y = \frac{1}{2}x^2$ .

### CAPÍTULO 3. FUNÇÕES E GRÁFICOS

**Exemplo 31** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = -x^2$ . A Figura 3.7, nos mostra como é o gráfico da função  $f$  comparado com os gráficos das funções  $y = -\frac{1}{2}x^2$  e  $y = -2x^2$ .

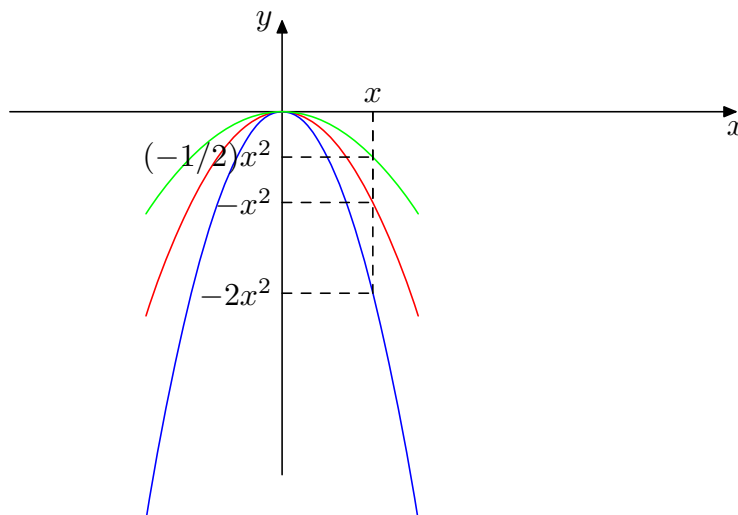


Figura 3.7: Comparação dos gráficos das funções  $y = -x^2$ ,  $y = -2x^2$  e  $y = -\frac{1}{2}x^2$ .

**Parábolas Transladadas:** Os exemplos 33, 34 e 35 mostram parábolas transladadas à partir da parábola  $y = x^2$ .

**Exemplo 32** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = x^2 + 2$ . A Figura 3.8, nos mostra como é o gráfico da função  $f$  comparado com o gráfico da função  $y = x^2$ .

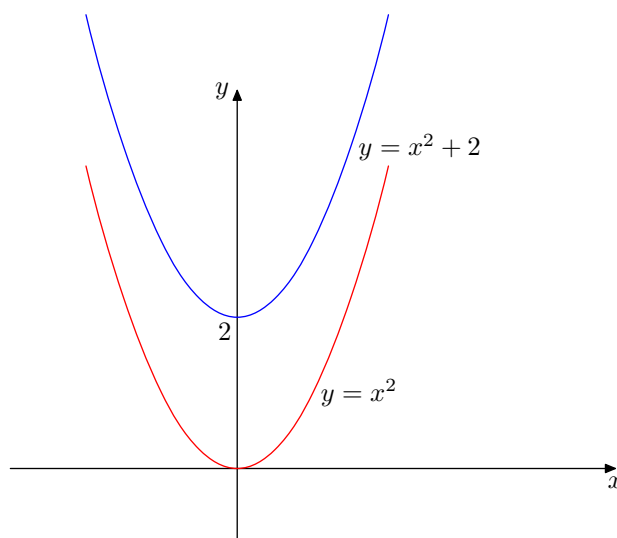


Figura 3.8: Comparação dos gráficos das funções  $y = x^2$  e  $y = x^2 + 2$ .



### CAPÍTULO 3. FUNÇÕES E GRÁFICOS

**Exemplo 33** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = x^2 - 3$ . A figura 3.9, nos mostra como é o gráfico da função  $f$  comparado com o gráfico da função  $y = x^2$ .

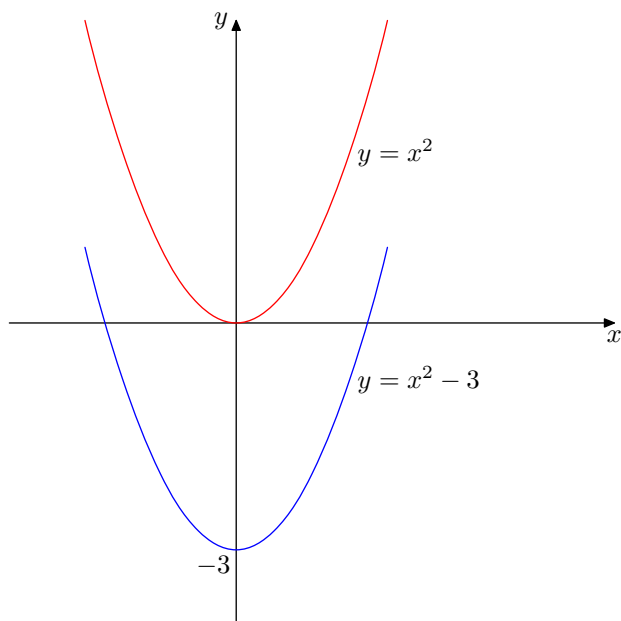


Figura 3.9: Comparação dos gráficos das funções  $y = x^2$  e  $y = x^2 - 3$ .

**Exemplo 34** Considere a função  $f : \mathbb{R} \rightarrow \mathbb{R}$ , definida por  $f(x) = (x - 3)^2$ . A figura 3.10, nos mostra como é o gráfico da função  $f$  comparado com o gráfico da função  $y = x^2$ .

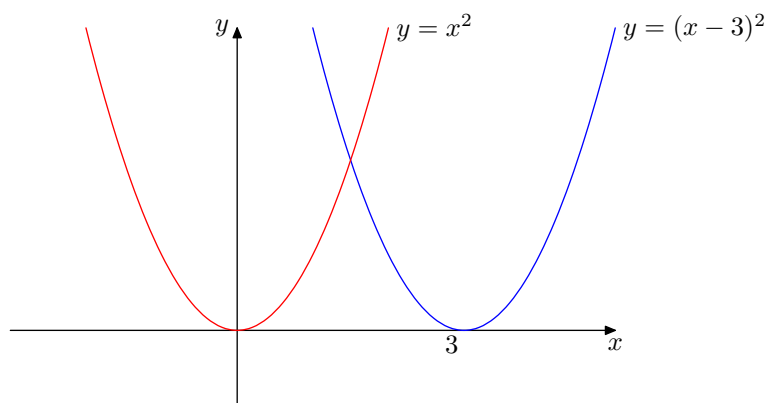


Figura 3.10: Comparação dos gráficos das funções  $y = x^2$  e  $y = (x - 3)^2$ .

### CAPÍTULO 3. FUNÇÕES E GRÁFICOS

**Caso geral:** Considere agora  $f(x) = ax^2 + bx + c$ , onde  $a \neq 0$ . Temos que

$$ax^2 + bx + c = a \left( x^2 + \frac{b}{a}x + \frac{c}{a} \right) = a \left[ \left( x + \frac{b}{2a} \right)^2 - \left( \frac{b}{2a} \right)^2 + \frac{c}{a} \right]$$

$$ax^2 + bx + c = a \left[ \left( x + \frac{b}{2a} \right)^2 - \frac{b^2 - 4ac}{4a^2} \right]$$

Indicando por  $\Delta = b^2 - 4ac$

$$f(x) = ax^2 + bx + c = a \left[ \left( x + \frac{b}{2a} \right)^2 \right] - \frac{\Delta}{4a}$$

A seguir na Figura 3.11, temos exemplos de alguns gráficos:

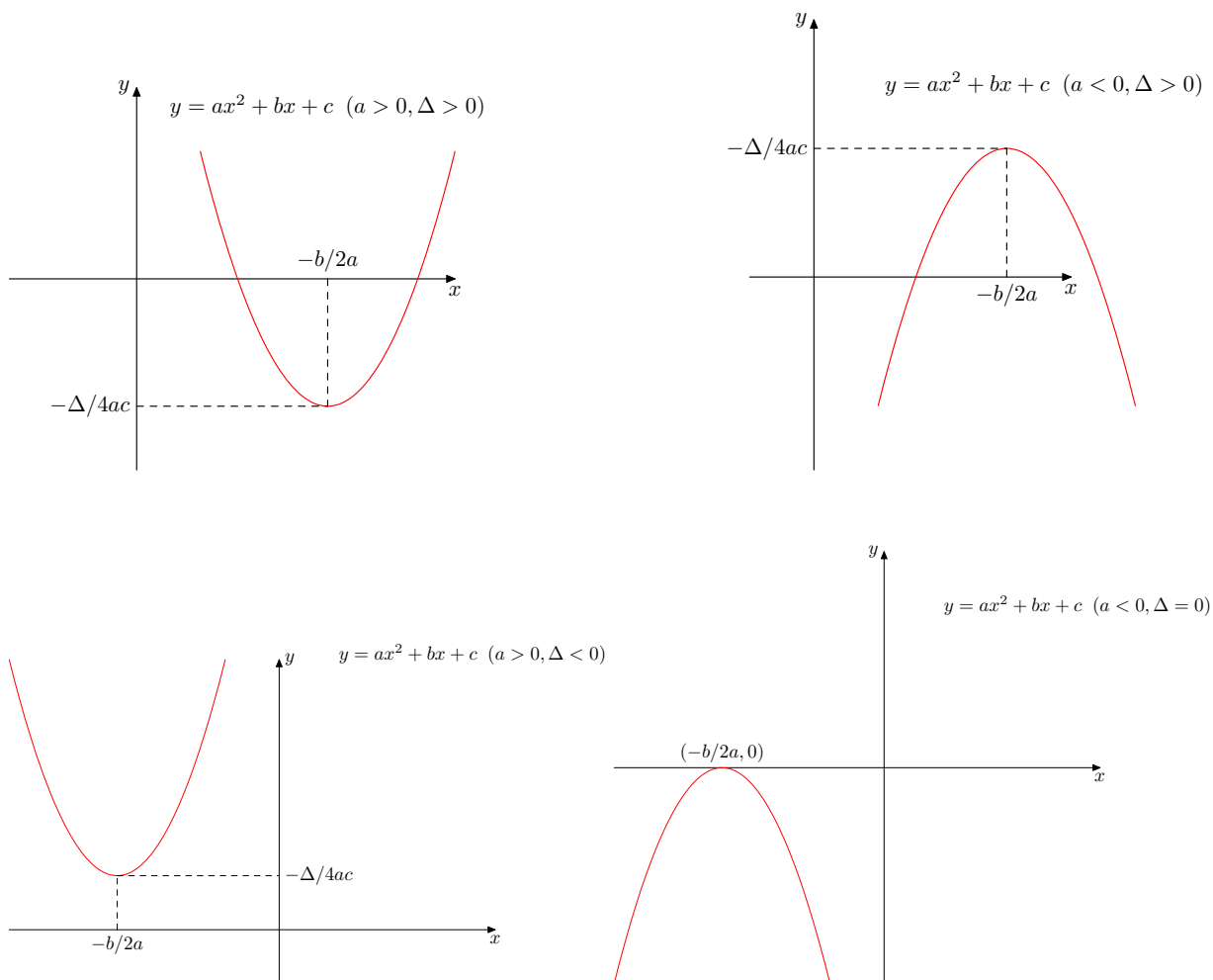


Figura 3.11: Gráficos de parábolas.

### 3.1.3 Funções polinomiais

Funções do tipo  $y = f(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} \dots + a_1 x + a_0$  onde  $a_n, a_{n-1}, \dots, a_1, a_0$  são números reais e  $n$  é um natural, com  $a_n \neq 0$ , são chamadas de funções polinomiais, de grau  $n$ .

**Exemplo 35** A função  $g : \mathbb{R} \rightarrow \mathbb{R}$  definida pela regra  $y = g(x) = x^3 - 3x - 2$  é uma função polinomial de grau 3. O gráfico de  $g$  é representado pela Figura 3.12.

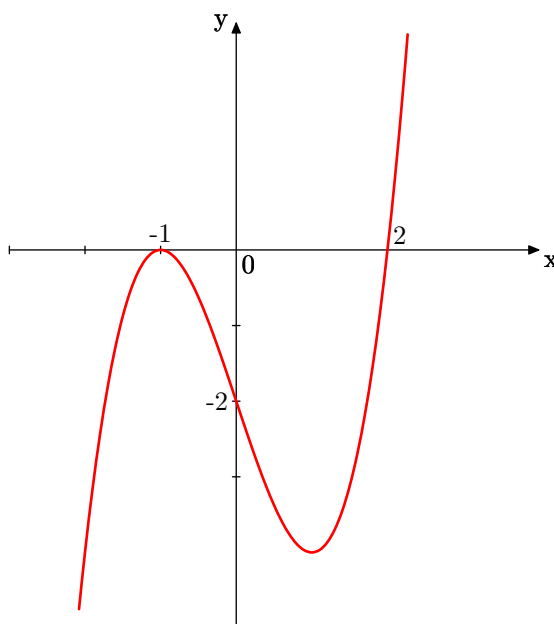


Figura 3.12: Gráfico da função polinomial  $y = g(x) = x^3 - 3x - 2$

## 3.2 Construção do Plano Cartesiano

Esboçar o gráfico de uma função  $f$  significa marcar todos os pontos  $(x, y)$  tais que  $y = f(x)$ . Também é necessário construir o plano cartesiano onde será representada a função. Na Figura 3.13 é mostrado o código de como desenhar esse plano limitado entre os intervalos  $[-4, 4]$  em  $x$  e  $[-4, 4]$  em  $y$ .

```

u:=0,7cm;
minx:=-4;maxx:=4;
miny:=-4;maxy:=4;
drawarrow ((minx, 0)--(maxx, 0)) scaled u;
drawarrow ((0, miny)--(0, maxy)) scaled u;
for i=minx*u step u until maxx*u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.lrt("x", (maxx*u, 0));
label.lft("y", (0, maxy*u));
label.llft("0", (0, 0));
endfor;

```

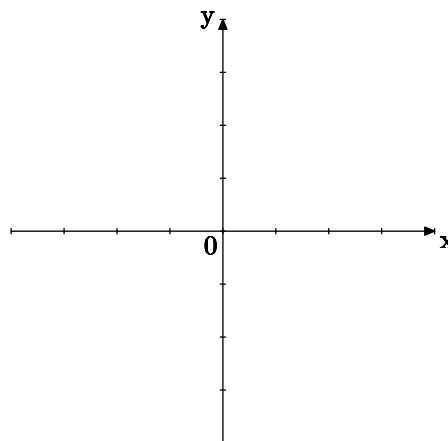


Figura 3.13: Plano cartesiano limitado pelo intervalo  $[-4, 4]$

### 3.3 Construção de gráficos de funções reais básicas

Para desenhar o gráfico de uma função real  $y = f(x)$  num intervalo  $[a, b]$  com o uso do METAPOST utiliza-se a seguinte técnica.

Inicialmente declara-se uma variável do tipo `path` para representar uma sequência de pontos desse gráfico. Após isso, o intervalo  $[a, b]$  deve ser dividido em  $n$  subintervalos de comprimento  $h$ . Assim, é necessário declarar as variáveis numéricas  $n$ ,  $a$ ,  $b$  e  $h$  onde  $a$  e  $b$  são as extremidades do intervalo  $[a, b]$  em que o gráfico será esboçado e  $h = \frac{b-a}{n}$ . A variável  $h$  é utilizada para obter uma sequência de abscissas, a partir de  $a$ . Em seguida, são definidas as coordenadas dos pontos que o METAPOST utilizará para traçar o desenho. Serão definidos  $n+1$  pontos do tipo  $z[i] = (x[i], y[i])$ , obtidos para  $i=0, 1, 2, \dots, n$ , cujas abscissas são da forma  $x[i] := a + i \cdot h$  e a respectiva ordenada  $y[i] := f(x)$ , onde  $f(x)$  representa a expressão da função a qual deseja construir o gráfico. Para traçar o gráfico,  $p[1]$  da função real  $y = f(x)$ , aplica-se o comando `draw p[1]`; onde

$$p[1] := z[0] .. z[1] .. z[2] .. \dots z[n];$$

Para valores altos de  $n$ , é conveniente utilizar o laço

$$p[1] := z[0] \text{ for } j=1 \text{ upto } n: \dots z[j] \text{ endfor};$$

Assim, nos exemplos a seguir, pode ser visto o código e a figura que representa o gráfico de uma função real definida e contínua num intervalo  $[a, b]$ . Nesses exemplos, o

## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

código que desenha o plano cartesiano utilizado será omitido, contendo assim somente os códigos do gráfico da função desejada e demais incrementos.

**Exemplo 36** A Figura 3.14 mostra o gráfico da função real  $y = f(x) = 2x$  no intervalo  $[-1, 1]$ .

```

path p[];n:=8;a:=-1;b:=1;h:=(b-a)/n;
for i=0 upto n: x[i]:=a+i*h;
y[i]:=x[i]*2; endfor
p[1]:=z[0] for j=1 upto n: ..z[j] endfor;
draw (u,2u)--(u,0) dashed evenly;
draw (u,2u)--(0,2u) dashed evenly;
label.bot("1", (u, 0));
label.lft("2", (0, 2u));
draw (-u,-2u)--(-u,0) dashed evenly;
draw (-u,-2u)--(0,-2u) dashed evenly;
label.top("-1", (-u, 0));
label.rt("-2", (0, -2u));
pickup pencircle scaled 1bp;
draw p[1] scaled u withcolor blue;

```

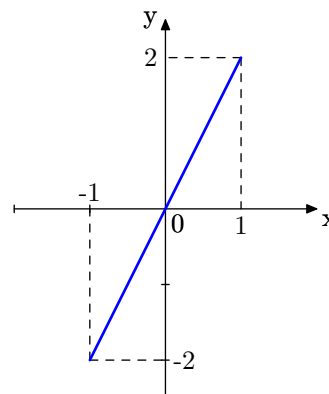


Figura 3.14: Gráfico da função real  $y = f(x) = 2x$  no intervalo  $[-1, 1]$ .

**Exemplo 37** Veja na Figura 3.15, o gráfico da função real  $y = f(x) = x^2 + 1$  no intervalo  $[-2, 2]$ .

```

path p[];n:=8;a:=-2;b:=2;h:=(b-a)/n;
for i=0 upto n: x[i]:=a+i*h;
y[i]:=-(x[i]**2)+1; endfor
p[1]:=z[0] for j=1 upto n: ..z[j] endfor;
draw (-2u,-3u)--(-2u,0) dashed evenly;
draw (-2u,-3u)--(0,-3u) dashed evenly;
label.top("-2", (-2u, 0));
label.top("2", (2u, 0));
label.ulft("-3", (0, -3u));
draw (2u,-3u)--(2u,0) dashed evenly;
draw (2u,-3u)--(0,-3u) dashed evenly;
label.ulft("1", (0, u));
pickup pencircle scaled 1 bp;
draw p[1] scaled u withcolor red;

```

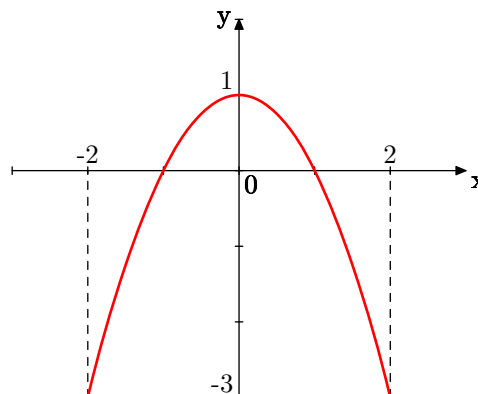


Figura 3.15: Gráfico da função real  $y = f(x) = x^2 + 1$  no intervalo  $[-2, 2]$ .

## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

**Exemplo 38** Na Figura 3.16 temos o gráfico da função polinomial  $y = f(x) = -x^4 + 10x^3 - 35x^2 + 50x - 24$  no intervalo  $]0, 5[$ .

```

path p[],q;n:=100;a:=0;b:=5;h:=(b-a)/n;
q:=(minx,miny)--(maxx, miny)--(maxx, maxy)
--(minx, maxy)--cycle) scaled 0.95;
for i=0 upto n: x[i]:=a+i*h;
y[i]:=-(x[i]**4)+10*(x[i]**3)
-35*(x[i]**2)+50*x[i]-24; endfor
p[1]:=z[0] for j=1 upto n: ..z[j] endfor;
label.urt("2", (2u, 0));
label.ulft("3", (3u, 0));
label.urt("4", (4u, 0));
label.ulft("1", (u, 0));
pickup pencircle scaled 1 bp;
draw ((p[1] cutafter q) cutbefore q)
scaled u withcolor green;

```

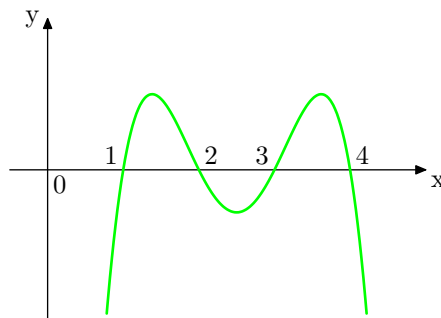


Figura 3.16: Gráfico da função polinomial  $y = f(x) = -x^4 + 10x^3 - 35x^2 + 50x - 24$  no intervalo  $]0, 5[$ .

Como ocorre no Exemplo 39, o gráfico de uma função pode extrapolar os limites definidos para os eixos  $x$  e  $y$ . Para cortar esse excesso, é definida uma região do plano (retangular) com os limites definidos para os eixo coordenados e armazenada na variável  $q$ . Sendo assim, com o comando `draw ((p[1] cutafter q) cutbefore q)` traça-se o gráfico delimitado por essa região.

**Exemplo 39** Gráfico da função real  $y = f(x) = 3x$  no intervalo  $[-1, 1]$ . Veja Figura 3.17.

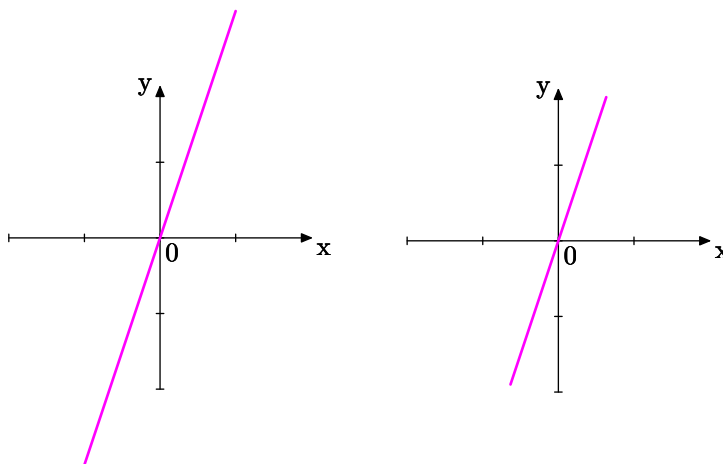


Figura 3.17: Gráficos da função real  $y = f(x) = 3x$  no intervalo  $[-1, 1]$ , extrapolado (à esquerda) e cortado (à direita).

## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

Uma vantagem muito interessante do METAPOST é que ele consegue calcular e identificar pontos de intersecção entre figuras. Uma das maneiras de se fazer isso é utilizando uma operação primitiva chamada `intersectiontimes`.

Esta (operação binária secundária) é uma das várias operações que lidam com os caminhos parametricamente. Ele localiza uma intersecção entre dois caminhos, fornecendo o parâmetro de “tempo” em cada caminho, onde  $t$  varia de zero ao número  $n$  de segmentos curvos que foram divididos o intervalo  $[a, b]$  onde foi construído um caminho.

Assim, dados dois caminhos  $p[i]$  e  $p[j]$  definidos pelas variáveis do tipo `path`, o comando

```
(t,whatever) = p[i] intersectiontimes p[j];
```

fornece o parâmetro de “tempo”  $t$  que define a localização do ponto de intersecção entre esses caminhos. Com esse parâmetro, é possível armazenar a localização do ponto numa variável do tipo `pair` fazendo  $z[n] = \text{point } t \text{ of } p[i]$  ou  $z[n] = \text{point } t \text{ of } p[j]$  já que o ponto de intersecção é comum aos dois caminhos e  $n$  é um índice ainda não utilizado.

Essa aplicação é utilizada no Exemplo 40.

**Exemplo 40** *Determinar o ponto de intersecção entre as curvas de equações  $y = 2^x$  e  $y = -x + 2$  e marcá-lo, conforme Figura 3.18.*

```
numeric u,h,n;u:=1.0cm;path p[],eixox,eixoy;
eixox:=(-2u,0)--(4u,0);
eixoy:=(0,-2u)--(0,4u);
drawarrow eixox;label.bot("x",(4u,0));
drawarrow eixoy;label.lft("y",(0,4u));
n:=10;a:=-2;b:=2;h:=(b-a)/n;
for i=0 upto n:x[i]:=a+i*h;
y[i]:=2**x[i]; w[i]:=-x[i]+2;
endfor;j:=0;p[1]=(x[j]*u,y[j]*u)%
for j=1 upto n: ..(x[j]*u,y[j]*u) endfor;
draw p[1] withcolor blue;
p[2]=(x[j]*u,w[j]*u)%
for j=1 upto n: ..(x[j]*u,w[j]*u) endfor;
draw p[2] withcolor red;
(t,whatever)=p[1] intersectiontimes p[2];
z20=point t of p[1];dotlabel.rt("I",z20);
label.rt(btex $y=2^x$ etex,point n of p[1])
withcolor blue;
label.urt("y=-x+3",point n of p[2])
withcolor red;
```

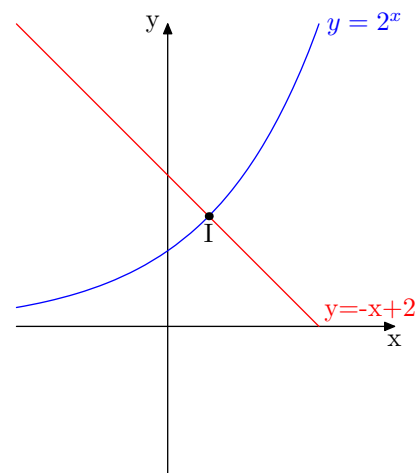


Figura 3.18: Intersecção I entre curvas no plano.

## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

Se existirem mais do que um ponto de intersecção entre os caminhos pode-se definir qual ponto que se deseja encontrar estabelecendo um intervalo específico do caminho  $p[i]$  através da operação `subpath`. Para tanto, deve-se armazenar numa variável  $p[k]$ , o “pedaço” do caminho a ser verificado, com o comando

$$p[k] = \text{subpath}(t_1, t_2) \text{ of } p[i]$$

e depois aplicar a operação

$$(t, \text{whatever}) = p[k] \text{ intersectiontimes } p[j];$$

onde  $p[j]$  é o outro caminho, como mostra o Exemplo 41.

**Exemplo 41** *Determinar um ponto de intersecção entre as curvas de equações  $y = 3x^2$  e  $y = -x + 2$  e marcá-lo, conforme Figura 3.19.*

```

q:=((-2u,-2u)--(3u,-2u)--(3u,4u)
--(-2u,4u)--cycle);
n:=10;a:=-2;b:=2;h:=(b-a)/n;
for i=0 upto n:
x[i]:=a+i*h;y[i]:=3*x[i]*x[i];w[i]:=-x[i]+2;
endfor; j:=0;
p[1]=(x[j]*u,y[j]*u)%
for j=1 upto n: ..(x[j]*u,y[j]*u) endfor;
draw ((p[1] cutafter q) cutbefore q)
withcolor blue + red;
p[2]=(x[j]*u,w[j]*u)%
for j=1 upto n: ..(x[j]*u,w[j]*u) endfor;
draw p[2] withcolor red + green;
p[3] =subpath(3,10) of p[2];
(t,whatever)=p[1] intersectiontimes p[3];
z20=point t of p[1];
dotlabel.rt("Q",z20);
label.rt(btex $y=3*x^2$ etex,point 8 of p[1])
withcolor blue + red;
label.urt("y=-x+2",point n of p[2])
withcolor red + green;

```

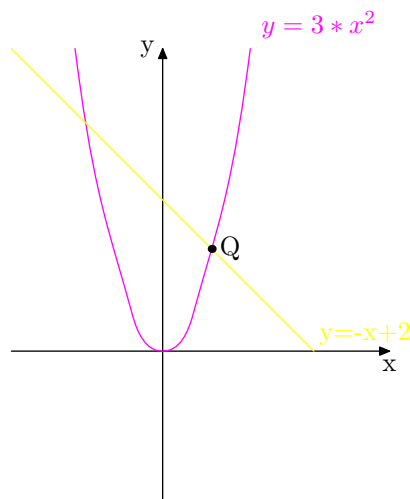


Figura 3.19: Intersecção  $Q$  entre curvas no plano.

Uma característica importante do METAPOST é a capacidade de resolver equações lineares. Por exemplo, o interpretador METAPOST pode ler  $a+b=3$ ; e  $2a=b+3$ ; e deduzir que  $a=2$  e  $b=1$ . Com isso, o METAPOST também pode encontrar o ponto de intersecção entre retas resolvendo o sistema formado pelas equações das mesmas, como é mostrado no Exemplo 42.



## CAPÍTULO 3. FUNÇÕES E GRÁFICOS

**Exemplo 42** Determinar o ponto de intersecção entre as retas de equações  $y=2x-1$  e  $y=-x+3$  e marcá-lo, conforme Figura 3.20.

```

numeric u,h,n,r,c,d;path p[],eixox, eixoy;
pair a[];u :=1 cm;
eixox=(-5u,0)--(4u,0);eixoy=(0,-4u)--(0,4u);
drawarrow eixox;drawarrow eixoy;
label.bot(btex $$ etex, (4u,0));
label.lft(btex $$ etex, (0,4u));
n:=20;a:=-1;b:=2;h:=(b-a)/n;
for i=0 upto n:
x[i]:=a+i*h;y[i]:=2*x[i];w[i]:=-x[i]+3;
endfor;
j:=0; p[1]=(x[j]*u,y[j]*u)
for j=1 upto n: ..(x[j]*u,y[j]*u) endfor;
draw p[1] withcolor blue;
p[2]=(x[j]*u,w[j]*u)
for j=1 upto n: ..(x[j]*u,w[j]*u) endfor;
draw p[2] withcolor red;
d:=-c+3=2c;
label.rt(btex $y=2x $ etex, (2u,4.2u));
label.lft(btex $y=-x+3$ etex, (3u,.7u));
label(btex $\bullet$ etex, (c,d)*u);

```

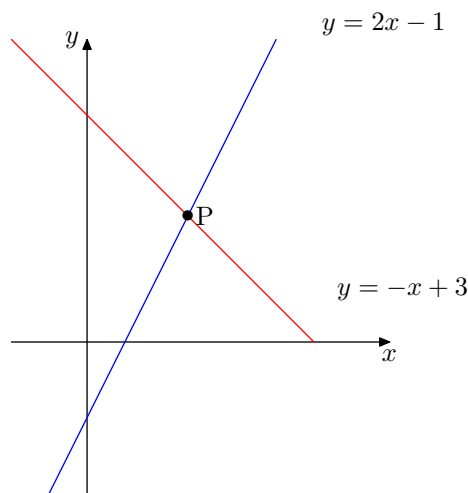


Figura 3.20: Intersecção P entre retas no plano.

**Exemplo 43** Determinar um ponto de intersecção entre as curvas de equações  $y = \sqrt{x}$  e  $y = x^2$  e marcá-lo, conforme Figura 3.21.

```

numeric u,h,n;path p[],eixox,eixoy;u=1.5cm;
eixox=(-u,0)--(2u,0);eixoy=(0,0)--(0,2u);
drawarrow eixox;drawarrow eixoy;
label.bot(btex ${x}$ etex,(2u,0));
label.lft(btex ${y}$ etex,(0,2u));
n:=10;a:=0;b:=2;h:=(b-a)/n;
for i=0 upto n:x[i]:=a+i*h;
y[i]:=x[i]*x[i]/2;w[i]:=sqrt(x[i]);endfor;
j:=0;p[1]=(x[j]*u,y[j]*u)%
for j=1 upto n: ..(x[j]*u,y[j]*u) endfor;
draw p[1] withcolor blue;p[2]=(x[j]*u,w[j]*u)%
for j=1 upto n: ..(x[j]*u,w[j]*u) endfor;
draw p[2] withcolor red;p[3]=subpath (2,10)of p[2];
(t,whatever)=p[1] intersectiontimes p[3];
z20=point t of p[1];label(btex $\bullet$ etex,z20);
label.rt(btex $y=x^2$ etex,point n of p[1])
withcolor blue;
label.rt(btex $y=\sqrt{x}$ etex,point n of p[2])
withcolor red;label.ulft("T",z20);

```

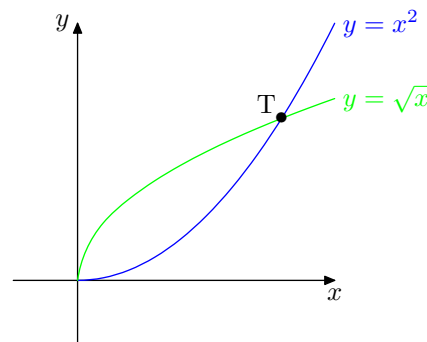


Figura 3.21: Intersecção I entre as curvas de equações  $y = \sqrt{x}$  e  $y = x^2$ .

# Capítulo 4

## Transformações com o METAPOST

Neste capítulo, estudaremos os comandos do METAPOST que possibilitam efetuar transformações de coordenadas no plano como translação, rotação e homotetia, obtendo assim, figuras com posições e tamanhos diferentes de outras pré desenhadas. Referências bibliográficas: HENDERSON [10], MUNIZ NETO[1], HOBBY [3] e HECK [11].

### 4.1 Transformação no Plano Cartesiano

**Definição 2** Uma transformação no plano  $\mathbb{R}^2$  é uma função  $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  que associa a cada ponto  $P$  de  $\mathbb{R}^2$  um ponto  $P' = T(P)$ . O ponto  $P'$  é a imagem do ponto  $P$  pela função  $T$ .

**Exemplo 44** Seja  $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  definida por  $T(x, y) = (x, y)$ . Essa função transforma, por exemplo, o ponto  $A=(1,2)$  no ponto  $A'=(1,2)$  e é chamada de Transformação identidade. No METAPOST essa transformação é aplicada por meio do comando *identity*.

**Exemplo 45** Seja  $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  definida por  $T(x, y) = (y, x)$ . Essa função transforma, por exemplo, o ponto  $B=(-2,3)$  no ponto  $B'=(3,-2)$ , ou seja, ela inverte as coordenadas de qualquer ponto  $(x,y)$ . Aplicada nos pontos do gráfico da função  $f : \mathbb{R} \rightarrow \mathbb{R}$  definida pela regra  $f(x) = 4x$  essa transformação  $T$  gera o gráfico da função  $f^{-1} : \mathbb{R} \rightarrow \mathbb{R}$ , chamada inversa de  $f$  e definida pela regra  $f^{-1}(x) = \frac{x}{4}$ , como é representado pela Figura 4.1.

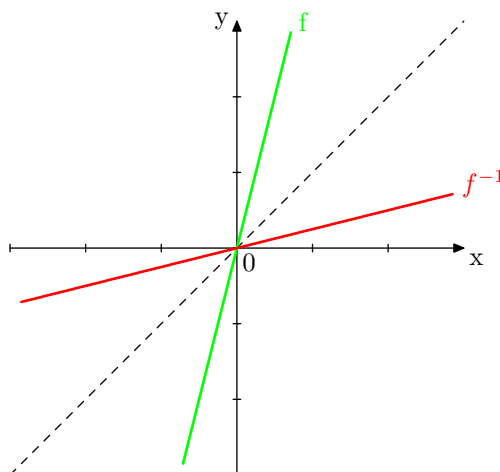


Figura 4.1: Gráfico da função  $f$  e da função  $f^{-1}$  obtida de  $f$  usando a transformação  $T(x, y) = (y, x)$

Como visto na Seção 2.9.5, as transformações no  $\mathbb{R}^2$  são aplicadas, no METAPOST usando o comando `transform`. Esse comando é utilizado para fazer alterações em figuras como translações, rotações e redimensionamento. Para tanto, é necessário aplicá-lo a pontos ou em figuras armazenadas, previamente, em variáveis do tipo `path`.

Para utilizar esse comando, inicialmente declara-se a transformação `T`, escrevendo `transform T`; e em seguida define-se a variável `T` precedida da transformação `identity`, utilizada para estabelecer um ponto de partida para demais transformações.

```
T:= identity <transformação predefinida>;
```

Dentre várias transformações predefinidas pelo METAPOST neste trabalho será estudado apenas as transformações básicas, utilizadas para realizar redimensionamento, rotações e translações em figuras do  $\mathbb{R}^2$ .

## 4.2 Homotetia

Considere a transformação  $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ , definida por  $T(x, y) = (ax, ay)$ , onde  $a$  é um número real positivo. Esta transformação é denominada homotetia de centro na origem e razão  $a$ . Esta homotetia aplicada a uma figura diminui ( $a < 1$ ) ou aumenta ( $a > 1$ ) o tamanho da figura.

O comando usado no METAPOST para aplicar homotetia de centro na origem e razão  $a$  é o `scaled a`. Assim, o comando `scaled a` representa  $T(x, y) = (ax, ay)$ .

**Exemplo 46** *Seja o triângulo cujos vértices são  $A=(0,0)$ ,  $B=(2,5)$  e  $C=(3,1)$  e considere a transformação  $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  definida por  $T(x, y) = (\frac{3}{4}x, \frac{3}{4}y)$ . Aplicando a transformação  $T$  no triângulo  $\triangle ABC$ , obtemos o triângulo  $\triangle ADE$ , como é representado pela Figura 4.2.*

```

path p,q;
pair A,B,C,D,E;
A:=(0,0);B:=(2u,3u);C:=(3u,u);
p:= A--B--C--cycle;
draw p;
label.llft("A", (0, 0));
label.top("B", (2u, 3u));
label.rt("C", (3u, u));
transform T;
T:= identity scaled 0.75;
q:= p transformed T;
pickup pencircle scaled 1 bp;
draw q withcolor red;
D:=(1.5u,2.25u);E:=(2.25u,0.75u);
label.ulft("D", (1.5u, 2.25u));
label.bot("E", (2.25u, 0.75u));

```

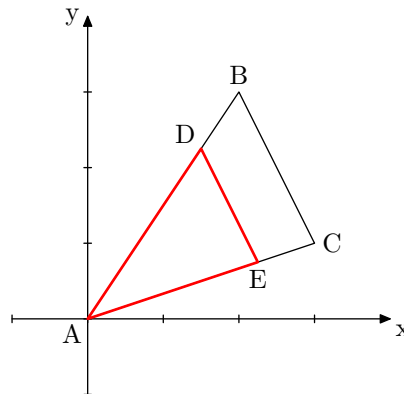


Figura 4.2: Homotetia no triângulo  $\triangle ABC$

A transformação indicada por `yscaled a` é a transformação  $T(x, y) = (x, ay)$ . Assim,  $(x, y)$  `yscaled a`  $\rightarrow (x, ay)$  preserva a abscissa e comprime ou expande a ordenada. Da mesma maneira, a transformação indicada por `xscaled a` é a transformação  $T(x, y) = (ax, y)$ . Assim,  $(x, y)$  `xscaled a`  $\rightarrow (ax, y)$  preserva a ordenada e comprime ou expande a abscissa.

### 4.3 Translação

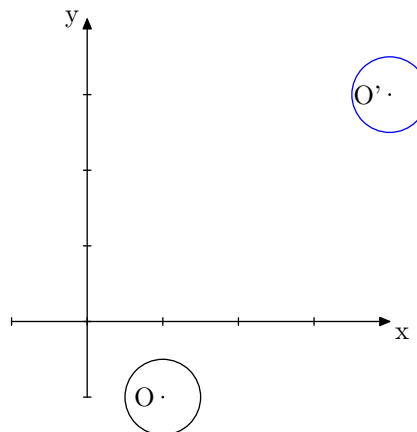
A transformação definida por  $T(x, y) = (x + a, y + b)$  é denominada translação. Uma translação aplicada a uma figura a desloca sem deformações.

Para utilizar a translação no METAPOST basta aplicar o comando `shifted`. Assim, o comando `shifted (a,b)` representa a transformação  $T(x, y) = (x + a, y + b)$ .

**Exemplo 47** *Seja a circunferência  $C_1$  de centro  $O=(u,-u)$  e raio 1 e considere a transformação  $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  definida por  $T(x, y) = (x + 3, y + 4)$ . Aplicando a transformação  $T$  na circunferência  $C$ , obtém-se a circunferência  $C_2$  de centro  $O'=(4,3)$  e raio 1, como é representado pela Figura 4.3.*

```

path p,q;
p:= fullcircle scaled u shifted (u,-u);
draw p;
transform T;
T:= identity shifted (3u,4u);
q:= p transformed T;
draw q withcolor blue;
pickup pencircle scaled 1bp;
drawdot (u,-u);
drawdot (u+3u,-u+4u);
label.lft("O", (u, -u));
label.lft("O'", (u+3u, -u+4u));
    
```


 Figura 4.3: Translação na circunferência  $C_1$ 

## 4.4 Rotação

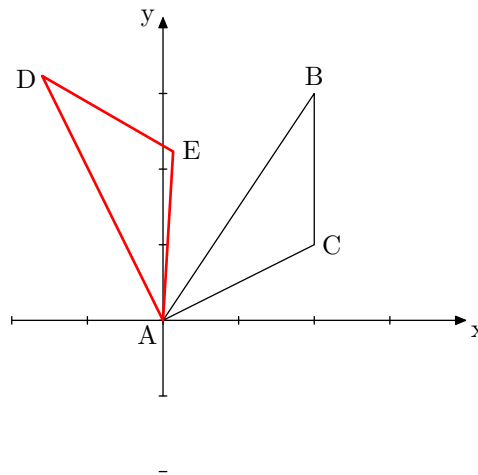
A rotação de uma figura é obtida quando fixado um ponto do plano e gira-se a figura de um ângulo  $\alpha$  qualquer, ao redor deste ponto. No  $\mathbb{R}^2$ , a rotação de um ponto de coordenadas  $(x, y)$  em torno da origem, é obtido através da transformação  $T(x, y) = (x \cdot \cos \alpha - y \cdot \text{sen} \alpha, x \cdot \text{sen} \alpha + y \cdot \cos \alpha)$ .

Para utilizar a rotação no METAPOST basta aplicar o comando `rotated`. Dessa forma, o comando `rotated  $\alpha$`  representa a rotação de uma figura em torno da origem de um ângulo  $\alpha$  graus.

**Exemplo 48** *Seja o triângulo cujos vértices são  $A=(0,0)$ ,  $B=(2,3)$  e  $C=(2,1)$ . Considere girar o triângulo  $\triangle ABC$  um ângulo de  $60^\circ$  em torno da origem. Obtemos o triângulo  $\triangle ADE$ , como é representado pela Figura 4.4.*

```

path p,q;
pair A,B,C,D,E;
A:=(0,0);B:=(2u,3u);C:=(2u,u);
p:= A--B--C--cycle;
draw p;
label.llft("A", (0, 0));
label.top("B", (2u, 3u));
label.rt("C", (2u, u));
transform T;
T:= identity rotated 60;
q:= p transformed T;
pickup pencircle scaled 1 bp;
draw q withcolor red;
label.ulft("D", (-1.6u, 3u));
label.rt("E", (0.15u, 2.25u));
    
```


 Figura 4.4: Rotação de  $60^\circ$  em torno da origem no triângulo  $\triangle ABC$

## CAPÍTULO 4. TRANSFORMAÇÕES COM O METAPOST

Para rotacionar uma figura em torno de um ponto específico  $(x_0, y_0)$  deve-se aplicar o comando `rotatedaround`. Assim, `rotatedaround ((x_0, y_0), \alpha)` representa numa figura a rotação de um ângulo  $\alpha$  graus em torno do ponto de coordenadas  $(x_0, y_0)$ .

**Exemplo 49** *Seja o retângulo representado na cor verde, conforme Figura 4.5. Considere girar esse retângulo sob um ângulo de  $60^\circ$  em torno do ponto de coordenadas  $(2,2)$ . Será obtido o retângulo representado na cor cinza.*

```

u:= 1 cm; path F, G;transform T[];
pickup pencircle scaled 1bp;
drawarrow((-u,0)--(4u,0));drawarrow((0,-u)--(0,4u));
for i=0 step u until 3u:draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);endfor;
label.bot("x", (4u, 0));label.lft("y", (0, 4u));
label.llft("0", (0, 0));label.bot("2", (2u, 0));
label.lft("2", (0, 2u));F:=(0.5u,1.75u)--
(3.5u,1.75u)--(3.5u,2.25u)--(0.5u,2.25u)..cycle;
filldraw F withcolor green;
T2:= identity rotatedaround ((2u,2u),60) ;
G:= F transformed T2;filldraw G withcolor 0.8white;
dotlabel.urrt("P", (2u, 2u));
draw (0,2u)--(2u,2u) dashed evenly;
draw (2u,0)--(2u,2u) dashed evenly;

```

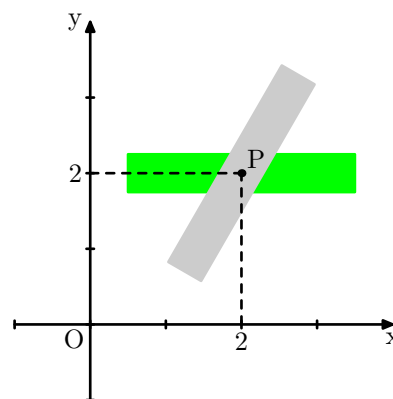


Figura 4.5: Rotação de  $60^\circ$  em torno do ponto  $(2,2)$ .

**Exemplo 50** *Seja o retângulo representado na cor amarelo, conforme Figura 4.6. Considere girar esse retângulo sob um ângulo de  $120^\circ$  em torno do ponto de coordenadas  $(1,2)$ . Será obtido o retângulo representado na cor cinza.*

```

u:= 1 cm; path F, G;transform T[];
pickup pencircle scaled 1bp;
drawarrow((-u,0)--(4u,0));drawarrow((0,-u)--(0,4u));
for i=-u step u until 3u:draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);endfor;
label.bot("x", (4u, 0));label.lft("y", (0, 4u));
label.llft("0", (0, 0));label.bot("1", (2u, 0));
label.lft("2", (0, 2u));F:=(0.5u,1.75u)--
(3.5u,1.75u)--(3.5u,2.25u)--(0.5u,2.25u)..cycle;
filldraw F withcolor green+red;
T2:= identity rotatedaround ((u,2u),90) ;
G:= F transformed T2;filldraw G withcolor 0.8white;
dotlabel.urrt("P", (u, 2u));
draw (0,2u)--(u,2u) dashed evenly;
draw (u,0)--(u,2u) dashed evenly;

```

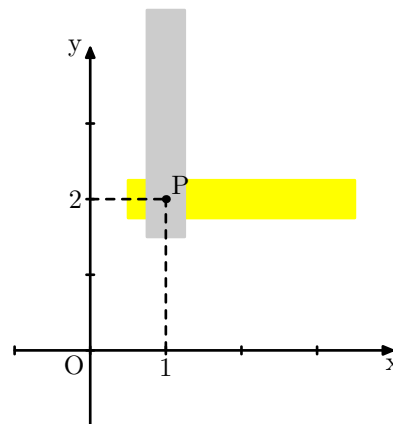


Figura 4.6: Rotação de  $120^\circ$  em torno do ponto  $(1,2)$ .

# Capítulo 5

## Construindo figuras com o METAPOST

Neste capítulo são apresentadas atividades, envolvendo a construção de figuras básicas utilizando o METAPOST, que podem ser realizadas com alunos do Ensino Médio, utilizando-se de computadores que possuem como programas instalados, o MIKTEX e um editor de textos T<sub>E</sub>X como o Texmaker. Referências bibliográficas: MELLO [14], EXEMPLES [12], HOBBY [3], MPREVIEW[2] e AUSUBEL [5].

### 5.1 Atividade 1: Construção de segmentos

Construção de segmentos de reta de extremidades definidas e indicações de comprimento.

- (a) Construir um segmento de reta  $\overline{MN}$  com extremidades nos pontos de coordenadas (1, 2) e (4, 6) utilizando o cm como unidade de medida e cuja espessura é 0,5 mm.
- (b) Construir um segmento de reta  $\overline{EF}$  tracejado com extremidades nos pontos de coordenadas (0, 0) e (2, 5) utilizando o cm como unidade de medida e cuja espessura é 0,3 mm.
- (c) Construir um segmento de reta  $\overline{CD}$  pontilhado com extremidades nos pontos de coordenadas (0, 4) e (4, 2) utilizando o cm como unidade de medida e espessura igual a 0,7 mm.

Solução Esperada

Solução (a):

```
u=1cm;
pickup pencircle scaled 0.5mm ;
draw (1u,2u)--(4u,6u);
label.lft("M", (1u,2u));
label.rt("N", (4u,6u));
label.ulft(btex $5\sim\text{cm}$ etex, (2.5u,4u));
```

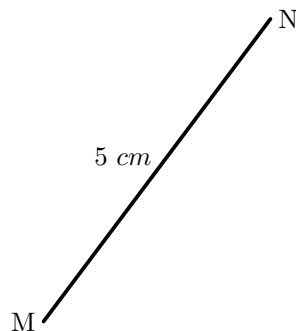


Figura 5.1: Segmento  $\overline{MN}$  de medida 5 cm

Solução (b):

```
u=1cm;
pickup pencircle scaled 0.3mm;
draw (0,0)--(2u,3u)evenly;
dotlabel.lft("E", (0,0));
dotlabel.rt("F", (2u,3u));
label.ulft(btex $\sqrt{13}\sim\text{cm}$ etex, (u,1.5u));
```

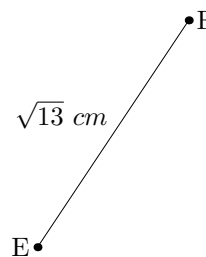


Figura 5.2: Segmento tracejado  $\overline{EF}$  de medida  $\sqrt{13}$  cm

Solução (c):

```
u=1cm;
pickup pencircle scaled 0.7mm ;
draw (0,4u)--(4u,2u)whittdots;
dotlabel.lft("C", (0,4u));
dotlabel.rt("D", (4u,2u));
label.urc(btex $\sqrt{20}\sim\text{cm}$ etex, (2u,3u));
```

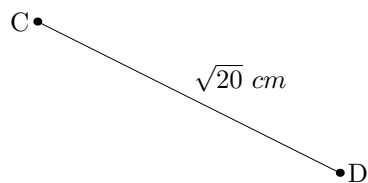


Figura 5.3: Segmento pontilhado  $\overline{CD}$  de medida  $\sqrt{20}$  cm



## 5.2 Atividade 2: Construção de eixos

Construção de eixos orientados e sistemas de eixos ortogonais.

- (a) Construir um eixo horizontal, com orientação positiva para a direita, utilizando o cm como unidade de medida e cuja espessura é 0,5 mm.
- (b) Construir um eixo vertical, com orientação positiva para a cima, utilizando o cm como unidade de medida e cuja espessura é 0,5 mm.
- (c) Construir um sistema de eixos ortogonais, utilizando o cm como unidade de medida e cuja espessura é 0,5 mm.

### Solução Esperada

Solução (a):

```

u:=1cm;
drawarrow (-4u,0)--(4u,0);
for i=-3u step u until 3u:
draw (i,u/20)--(i,-u/20);
endfor;
label.bot("x", (4u, 0));
label.bot("0", (0, 0));
    
```

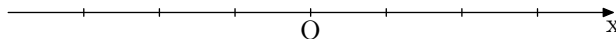


Figura 5.4: Eixo horizontal Ox

Solução (b):

```

u:= 1 cm;
drawarrow ((0, -3u)--(0, 3u));
for i=-2u step u until 2u:
draw (u/20,i)--(-u/20,i);
label.lft("y", (0, 3u));
label.lft("0", (0, 0));
endfor;
    
```

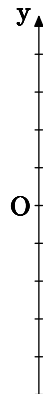


Figura 5.5: Eixo vertical Oy

Solução (c):

```

u:= 1 cm;
drawarrow ((-4u, 0)--(4u, 0));
drawarrow ((0, -4u)--(0, 4u));
for i=-4u step u until 4u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.bot("x", (4u, 0));
label.lft("y", (0, 4u));
label.ulft("O", (0, 0));
endfor;

```

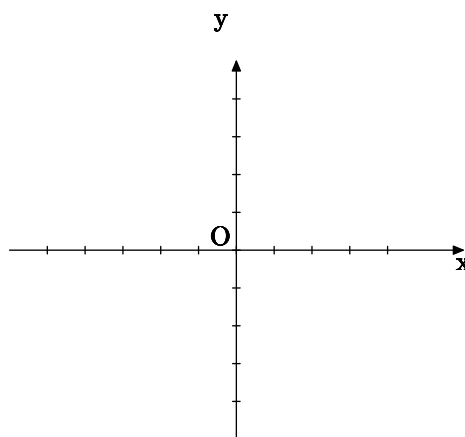


Figura 5.6: Sistema de coordenadas cartesianas ortogonais Oxy

### 5.3 Atividade 3: Construção de ângulos

Construção de ângulos com medidas definidas.

- (a) Construir um ângulo  $A\hat{O}B$  de medida  $\alpha$  igual a 30 graus.
- (b) Construir um ângulo  $P\hat{Q}R$  de medida  $\theta$  igual a 75 graus.
- (c) Construir um ângulo  $X\hat{Y}Z$  de medida  $\beta$  igual a 140 graus.

#### Solução Esperada

Solução (a):

```

u:=1cm;
alpha := 30;
draw (0,0) -- 2u*dir 0;
draw (0,0) -- 2u*dir alpha;
draw u * dir 0 {dir 90} ..u * dir alpha {dir(90+alpha)};
label.lft("O", origin);
label.bot("A", (u, 0));
label.ulft("B", u*dir alpha);
label(btex $\alpha=30^{\circ}$ etex, 1.8cm*dir (alpha/2));

```

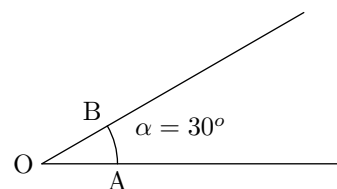


Figura 5.7: Ângulo  $A\hat{O}B$  de medida  $\alpha = 30^\circ$ .

**Solução (b):**

```

u:=1cm;
teta := 75;
draw (0,0) -- 4u*dir 0;
draw (0,0) -- 4u*dir teta;
draw u * dir 0 {dir 90}..u * dir teta {dir(90+teta)};
label.llft("Q", origin);
label.bot("P", (u, 0));
label.ulft("R", u*dir teta);
label(btex $\theta=75^\circ$ etex, 1.9cm*dir (teta/2));

```

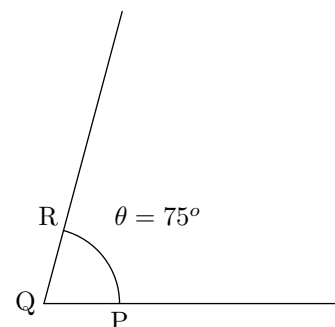


Figura 5.8: Ângulo  $P\hat{Q}R$  de medida  $\theta = 75^\circ$ .

**Solução (c):**

```

u:=1cm;
beta := 140;
draw (0,0) -- 2u*dir 0;
draw (0,0) -- 2u*dir beta;
draw u * dir 0 {dir 90}..u * dir beta {dir(90+beta)};
label.llft("Y", origin);
label.bot("X", (u, 0));
label.llft("Z", u*dir beta);
label(btex $\beta=140^\circ$ etex, 1.7cm*dir (beta/2));

```

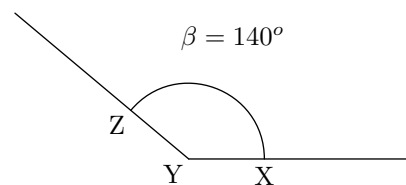


Figura 5.9: Ângulo  $X\hat{Y}Z$  de medida  $\beta = 140^\circ$ .

## 5.4 Atividade 4: Construção de circunferências

Construção de circunferências com raios e centros determinados num sistema de eixos ortogonais.

- (a) Construir uma circunferência com centro na origem do plano cartesiano e raio igual a 2 cm e cuja espessura é 0,6 mm.
- (b) Construir uma circunferência com centro no ponto de coordenadas (2, 1) do plano cartesiano e raio igual a 3, utilizando o cm como unidade, e cuja espessura é 0,6 mm na cor vermelha.
- (c) Construir uma circunferência com preenchimento na cor verde, centro no ponto de coordenadas (1, 1) do plano cartesiano e raio igual a 1, utilizando o cm como unidade, e cuja espessura é 0,7 mm.

### Solução Esperada

Solução (a):

```

u:= 1 cm;
drawarrow ((-3u, 0)--(3u, 0));
drawarrow ((0, -2.5u)--(0, 2.5u));
for i=-2.5u step .5u until 2.5u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.bot("x", (3u, 0));
label.lft("y", (0, 2.5u));
label.ulft("O", (0, 0));
endfor;
pickup pencircle scaled 0.6mm;
draw fullcircle scaled 2cm;
    
```

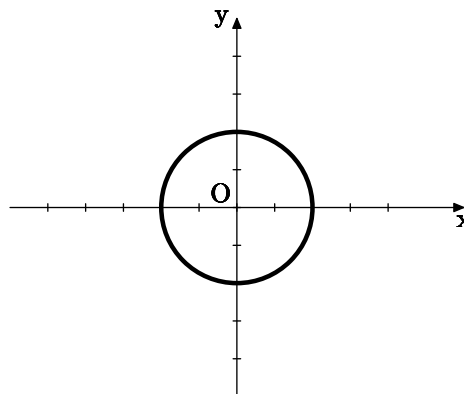


Figura 5.10: Circunferência de centro na origem e raio 2cm

Solução (b):

```

u:= 1 cm; path C;transform T;
drawarrow ((-2u, 0)--(4u, 0));
drawarrow ((0, -2u)--(0, 4u));
for i=-2u step u until 4u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.bot("x", (4u, 0));
label.lft("y", (0, 4u));
label.llft("O", (0, 0));
endfor;
T:= identity scaled 3cm shifted (2u, u);
C:= fullcircle transformed T;
draw (2u, u)--(2u, 0) dashed evenly;
draw (2u, u)--(0, u) dashed evenly;
pickup pencircle scaled 0.6mm;
draw C withcolor red;
pickup pencircle scaled 0.8mm;
drawdot (2u, u);
label.bot("2", (2u, 0));
label.lft("1", (0, u));
    
```

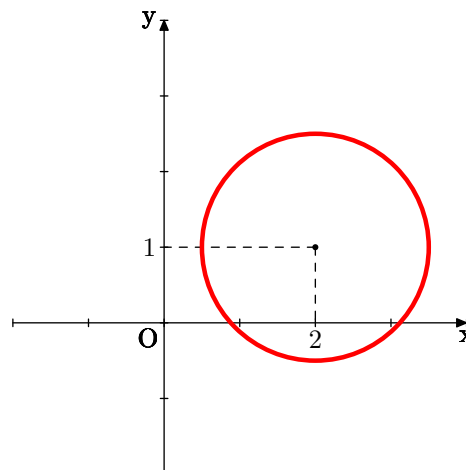


Figura 5.11: Circunferência de centro (2, 1) e raio 1,5 cm

Solução (c):

```

u:= 1 cm; path C;transform T;
T:= identity scaled 2cm shifted (u, u);
C:= fullcircle transformed T;
fill C withcolor green;
drawarrow ((-2u, 0)--(4u, 0));
drawarrow ((0, -2u)--(0, 4u));
for i=-2u step u until 3.5u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.bot("x", (4u, 0));
label.lft("y", (0, 4u));
label.llft("0", (0, 0));
endfor;
draw (u, u)--(u, 0) dashed evenly;
draw (u, u)--(0, u) dashed evenly;
pickup pencircle scaled 0.6mm; draw C;
pickup pencircle scaled 0.8mm;
drawdot (u, u);
label.bot("1", (u, 0));
label.lft("1", (0, u));

```

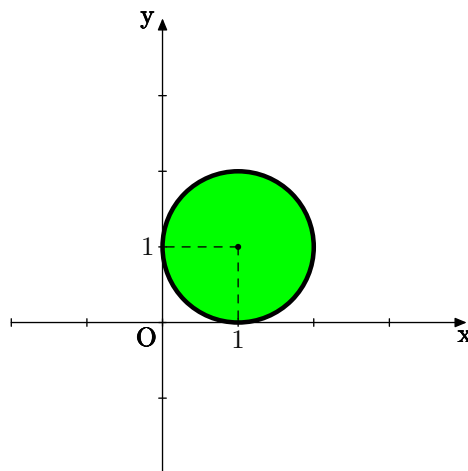


Figura 5.12: Circunferência de centro (1, 1) e raio 1 preenchida na cor verde

## 5.5 Atividade 5: Construção de triângulos

Construção de triângulos dada as suas dimensões e/ou seus ângulos internos.

- (a) Construir um triângulo  $\triangle ABC$  de lados medindo 3 cm, 5 cm e 6 cm com espessura de 0,5 mm.
- (b) Construir um triângulo  $\triangle PQR$  equilátero com 4 cm de lado.
- (c) Construir um triângulo  $\triangle MNO$  isósceles, cuja base mede 2 cm, seus lados congruentes medem 4 cm e seu interior preenchido com cor azul.

### Solução Esperada

#### Solução (a):

```

u:= 1 cm;path p[];pair A, B;
A:= origin;B:=(6u,0);
p[1]:= fullcircle scaled 6u shifted (0,0);
p[2]:= fullcircle scaled 10u shifted (6u,0);
(t,whatever)=p[1] intersectiontimes p[2];
z20=point t of p[1];label.top("C",z20);
draw A--B--z20--cycle;
label.llft("A",origin);label.lrt("B",(6u,0));
label.bot(btex $6\text{cm}$ etex, 0.5(A+B));
label.ulft(btex $3\text{cm}$ etex, 0.5(A+z20));
label.urrt(btex $5\text{cm}$ etex, 0.5(z20+B));
    
```

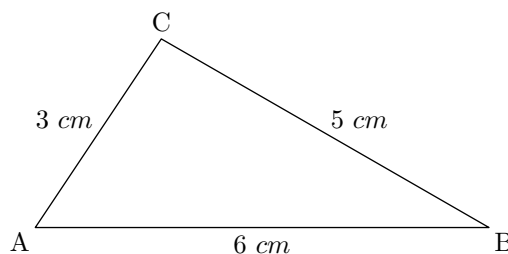


Figura 5.13: Triângulo  $\triangle ABC$  escaleno.

#### Solução (b):

```

u:= 1 cm;path p[]; pair P, Q;P:= (0,0); Q:=(4u,0);
p[1]:= fullcircle scaled 8u shifted (0,0);
p[2]:= fullcircle scaled 8u shifted (4u,0);
(t,whatever)=p[1] intersectiontimes p[2];
z20=point t of p[1];draw P--Q--z20--cycle;
label.top("R",z20);label.llft("P",origin);
label.lrt("Q",(4u,0));
label.bot(btex $4\text{cm}$ etex, 0.5(P+Q));
label.ulft(btex $4\text{cm}$ etex, 0.5(P+z20));
label.urrt(btex $4\text{cm}$ etex, 0.5(z20+Q));
    
```

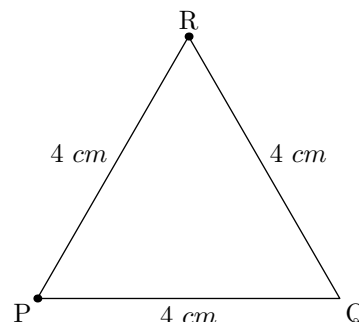


Figura 5.14: Triângulo  $\triangle PQR$  equilátero de lado 4 cm.

#### Solução (c):

```

u:= 1 cm;path p[]; pair M, N;M:= (0,0); N:=(2u,0);
p[1]:= fullcircle scaled 8u shifted (0,0);
p[2]:= fullcircle scaled 8u shifted (2u,0);
(t,whatever)=p[1] intersectiontimes p[2];
z20=point t of p[1];label.top("O",z20);
draw M--N--z20--cycle;fill M--N--z20--cycle withcolor blue;
label.llft("M", origin);label.lrt("N",N);
label.bot(TEX("$2\text{cm}$"), 0.5(M+N));
label.ulft(TEX("$4\text{cm}$"), 0.5(M+z20));
label.urrt(TEX("$4\text{cm}$"), 0.5(z20+N));
    
```

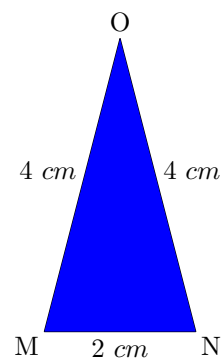


Figura 5.15: Triângulo  $\triangle MNO$  isósceles de base 2 cm

## 5.6 Atividade 6: Construção de gráfico de função afim

Construção de gráficos de função afim num sistema de eixos cartesianos ortogonais.

- (a) Construir o gráfico da função real  $y = x - 1$  no plano cartesiano e cuja espessura é 0,6 mm e cor azul.
- (b) Construir o gráfico da função real  $y = -2x$  no plano cartesiano e cuja espessura é 0,6 mm e cor vermelha.
- (c) Construir o gráfico da função real  $y = \frac{2}{3}x + 1$  no plano cartesiano e cuja espessura é 0,4 mm e cor verde.

### Solução Esperada

Solução (a):

```

u:= 1 cm; path C;transform T;
T:= identity scaled 3u shifted (u, u);
C:= (-2u,-3u)--(3u,2u);
pickup pencircle scaled 0.2mm;
drawarrow ((-3u, 0)--(3u, 0));
drawarrow ((0, -3u)--(0, 3u));
for i=-3u step u until 2.5u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.bot("x", (3u, 0));
label.lft("y", (0, 3u));
label.ulft("0", (0, 0));
endfor;
pickup pencircle scaled 0.4mm;
draw C withcolor blue;
label.lrt("1", (u, 0));
label.ulft("-1", (0, -u));
label.top("y=x-1", (3u, 2u)) withcolor blue;

```

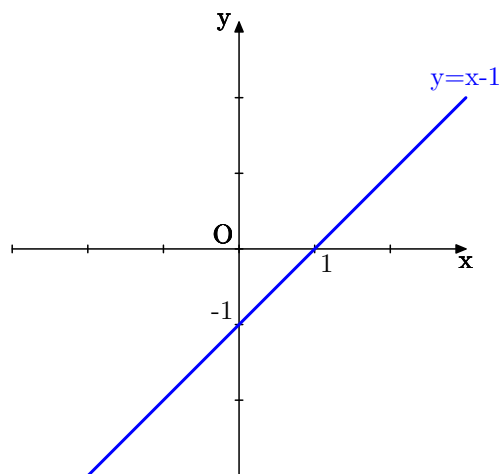


Figura 5.16: Gráfico da função real definida por  $y=x-1$  na cor azul

Solução (b):

```

u:= 1 cm; path C;transform T;
T:= identity scaled 3u shifted (u, u);
C:= (-1.5u,3u)--(1.5u,-3u);
pickup pencircle scaled 0.2mm;
drawarrow ((-3u, 0)--(3u, 0));
drawarrow ((0, -3u)--(0, 3u));
for i=-3u step u until 2.5u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.bot("x", (3u, 0));
label.lft("y", (0, 3u));
label.llft("0", (0, 0));
draw (u,-2u)--(u, 0) dashed evenly;
draw (u,-2u)--(0, -2u) dashed evenly;
endfor;
pickup pencircle scaled 0.4mm;
draw C withcolor red;
label.top("1", (u, 0));
label.lft("-2", (0, -2u));
label.top("y=-2x", (2u, -3u)) withcolor red;

```

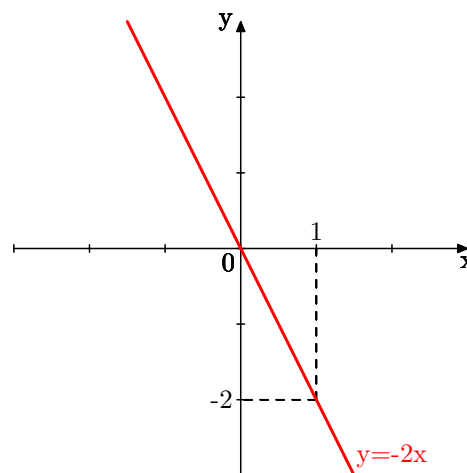


Figura 5.17: Gráfico da função real  $y=-2x$  na cor vermelha

Solução (c):

```

u:= 1 cm; path C;transform T;
T:= identity scaled 3u shifted (u, u);
C:= (-3u,-u)--(3u,3u);
pickup pencircle scaled 0.2mm;
drawarrow ((-3u, 0)--(3u, 0));
drawarrow ((0, -3u)--(0, 3u));
for i=-3u step u until 2.5u:
draw (i,u/20)--(i,-u/20);
draw (u/20,i)--(-u/20,i);
label.bot("x", (3u, 0));
label.lft("y", (0, 3u));
label.llft("0", (0, 0));
endfor;
pickup pencircle scaled 0.4mm;
draw C withcolor green;
label.bot(btex $-3/2$ etex, (-3u/2, 0));
label.ulft("1", (0, u));
label.bot(btex $y=\frac{2}{3}x+1$ etex,
(2.5u, 2u))
withcolor green;

```

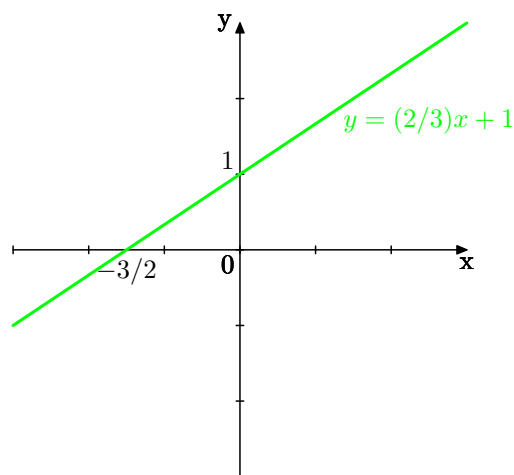


Figura 5.18: Gráfico da função real  $y = \frac{2}{3}x + 1$  na cor verde



## 5.7 Atividade 7: Construção de gráficos de funções quadráticas usando transformações.

Construção de gráficos de função quadrática num sistema de eixos cartesianos ortogonais com o uso de transformações.

- (a) Construir o gráfico da função real  $y = 2x^2$  no plano cartesiano, a partir da função  $y = x^2$ , usando uma transformação adequada, na cor azul.
- (b) Construir o gráfico da função real  $y = -2x^2 - 4x - 2 = -2(x + 1)^2$  à partir do gráfico da função  $y = x^2$ , na cor preta.
- (c) Aplicar uma rotação de  $30^\circ$  no sentido anti-horário ao gráfico da função  $f(x) = (x - 1)^2 + 2$  e desenhar o resultado na cor verde.

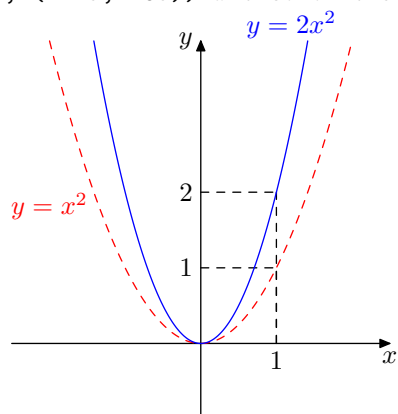
Solução Esperada

## Solução (a):

```

numeric u,h,n,r;path p[],f[], eiox, eioy, q;
u :=1 cm;eiox=(-2.5u,0)--(2.5u,0);
eioy=(0,-u)--(0,4u);
drawarrow eiox; drawarrow eioy;
label.bot(btex $x$ etex, (2.5u,0));
label.lft(btex $y$ etex, (0,4u));
q:= (-2.5u,-u)--(2.5u,-u)--(2.5u,4u)--(-2.5u,4u)--cycle;n:=100;a:=-2;b:=3;h:=(b-a)/n;
for i=0 upto n:x[i]:=a+i*h;y[i]:=x[i]*x[i];
endfor;
j:=0;p[1]=(x[j]*u,y[j]*u)
for j=1 upto n: ..(x[j]*u,y[j]*u) endfor;
draw ((p[1] cutafter q) cutbefore q)
withcolor red dashed evenly;
draw (u,0)..(u,2u) dashed evenly;
draw(0,2u)..(u,2u) dashed evenly;
draw (u,0)..(u,u) dashed evenly;
draw(0,u)..(u,u) dashed evenly;
label.bot(btex $1$ etex, (u,0));
label.lft(btex $1$ etex, (0,u));
label.lft(btex $2$ etex, (0,2u));
transform T[];
T1:= identity yscaled 2;
f1:= p1 transformed T1;
draw ((f1 cutafter q) cutbefore q)
withcolor blue;
label.bot(btex $y=x^2$ etex, (-2u,2.1u))
withcolor red;
label.bot(btex $y=2x^2$ etex, (1.2u,4.5u)) withcolor blue;

```

Figura 5.19: Gráfico da função  $y = 2x^2$  na cor azul

Solução (b):

```

numeric u,h,n,r;path p[],f[], eixox, eixoy, q;
u :=1 cm;eixox=(-2.5u,0)--(2.5u,0);
eixoy=(0,-3u)--(0,4u);
drawarrow eixox; drawarrow eixoy;
label.bot(btex $$ etex, (2.5u,0));
label.lft(btex $y$ etex, (0,4u));
q:=(-2.5u,-3u)--(2.5u,-3u)--(2.5u,4u)--(-2.5u,4u)--cycle;
n:=100;a:=-2;b:=3;h:=(b-a)/n;
for i=0 upto n:x[i]:=a+i*h;y[i]:=x[i]*x[i];
endfor;
j:=0;p[1]=(x[j]*u,y[j]*u)
for j=1 upto n: ..(x[j]*u,y[j]*u) endfor;
draw (-u,0)..(-u,u) dashed evenly;
draw(0,u)..(-u,u) dashed evenly;
label.bot(btex $-1$ etex, (-u,0));
label.rt(btex $1$ etex, (0,u));
transform T[];
T1:= identity yscaled -2;
f1:= p1 transformed T1;
T2:= identity shifted (-u,0);
f2:= f1 transformed T2;
draw ((p[1] cutafter q) cutbefore q)
withcolor red dashed evenly;
draw ((f[2] cutafter q) cutbefore q);
label.bot(btex $y=x^2$ etex, (-2u,2.1u))
withcolor red;

```

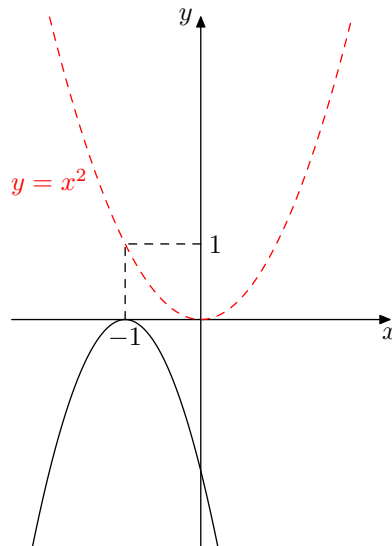


Figura 5.20: Gráfico da função  $f(x) = -2(x + 1)^2$ .

Solução (c):

```

numeric u,h,n,r;path p[],f[], eixox, eixoy, q;
u :=1 cm;eixox=(-2.5u,0)--(2.5u,0);
eixoy=(0,-3u)--(0,4u);
drawarrow eixox; drawarrow eixoy;
label.bot(btex  $x$  etex, (2.5u,0));
label.lft(btex  $y$  etex, (0,4u));
q:= (-2.5u,-3u)--(5u,-3u)--(5u,4u)--(-2.5u,4u)--cycle;
n:=100;a:=-2;b:=3;h:=(b-a)/n;
for i=0 upto n:x[i]:=a+i*h;y[i]:=x[i]*x[i];
endfor;
j:=0;p[1]=(x[j]*u,y[j]*u)
for j=1 upto n: ..(x[j]*u,y[j]*u) endfor;
draw ((p[1] cutafter q) cutbefore q)
withcolor red dashed evenly;
draw (-u,0)..(-u,u) dashed evenly;
draw(0,u)..(-u,u) dashed evenly;
label.bot(btex  $-1$  etex, (-u,0));
label.rt(btex  $1$  etex, (0,u));
transform T[];
T1:= identity yscaled 2;
f1:= p1 transformed T1;
T2:= identity shifted (u,-2u);
f2:= f1 transformed T2;
T3:= identity rotatedaround ((u,-2u),-30);
f3:= f2 transformed T3;
draw ((f3 cutafter q) cutbefore q)
withcolor green;
draw (u,0)..(u,-2u) dashed evenly;
draw(0,-2u)..(u,-2u) dashed evenly;
label.top(btex  $1$  etex, (u,0));
label.lft(btex  $-2$  etex, (0,-2u));
label.bot(btex  $y=x^2$  etex, (-2u,2.1u))
withcolor red;
    
```

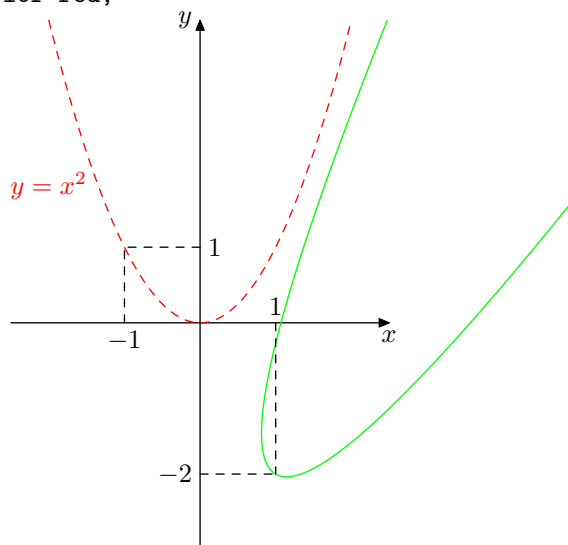


Figura 5.21: Gráfico do resultado da rotação de  $30^\circ$  no sentido anti-horário ao gráfico de  $f(x) = (x - 1)^2 + 2$ .

## 5.8 Relato da prática com os alunos

Neste capítulo, é apresentado um breve relato de como ocorreu o desenvolvimento deste trabalho, principalmente, a parte relacionada com a prática desenvolvida com os alunos.

Atualmente existe um grande desafio por parte de muitos docentes, em motivar os alunos no estudo da matemática. Tal dificuldade se dá à grande facilidade de encontrar informações prontas e rápidas sobre os diversos problemas encontrados no cotidiano, provocando assim, grande desinteresse na busca por explicações mais precisas e robustas desses problemas. As facilidades encontradas pelo avanço tecnológico não deixam espaço para os alunos desenvolverem pensamentos críticos e profundos a respeito do funcionamento do mundo que os rodeia e sobre a importância que as ciências, sobretudo a matemática, tem para explicar os fatos.

Nesse sentido, o trabalho apresenta uma proposta que motiva alunos e professores no processo de ensino-aprendizagem. Proposta essa que utiliza a programação para aprender matemática, através do uso do METAPOST.

O trabalho foi desenvolvido com alunos da turma 2º ano D do Ensino Médio Técnico em Programação de Jogos Digitais, período matutino, da Escola Estadual Hércules Maymone, no ano letivo de 2019, e foram utilizadas várias aulas do professor Marcelo Furquin.

A escolha para trabalhar com essa turma foi pela conveniência dos alunos já estarem aprendendo linguagens de programação para o desenvolvimento de jogos digitais, o que contribuiu significativamente para aprenderem a linguagem METAPOST. Esta ideia está baseada na teoria da aprendizagem significativa de David Ausubel que em seu livro Psicologia Educacional é contundente em dizer que “O fator isolado mais importante que influencia o aprendizado é aquilo que o aprendiz já conhece”. Portanto, aprender matemática, em particular, aprender geometria utilizando uma ferramenta que um grupo de alunos já domina previamente (a programação) foi uma boa estratégia para alcançar tal objetivo.

O trabalho se iniciou através de quatro aulas teóricas ministradas às sextas feiras, no horário de aula do professor Marcelo. Nessa aulas foram apresentados os comandos básicos sobre o METAPOST e como utilizá-lo na programação de figuras. Para isso foi elaborado um resumo com os comandos do METAPOST, em material impresso e entregue aos alunos.

## CAPÍTULO 5. CONSTRUINDO FIGURAS COM O METAPOST

A ideia era terminar essas aulas e iniciar a parte prática com desenvolvimento das figuras num computador e, posteriormente, propor uma série de atividades. Porém, alguns contratemplos ocorreram, como a precariedade dos computadores e a falta de internet no laboratório de informática da escola, e também, a mudança nos horários de aulas do professor Marcelo Furquim.

A ausência de internet acarretaria a impossibilidade de trabalhar na plataforma on-line do professor Troy Henderson (<http://www.tlhiv.org/mppreview/>). Essa plataforma é uma excelente ferramenta on-line que permite digitar o código numa caixa de texto e, num clique apenas, visualizar o resultado na tela.

A precariedade dos computadores não permitiu a instalação dos programas necessários para o desenvolvimento das figuras com o METAPOST, como o Miktex e o editor Texstudio ou o Texmaker.

Já a mudança de horários, ocorrida após uma troca na gestão da escola, gerou troca de muitos professores, impedindo assim que as aulas do professor Marcelo na turma em questão não coincidisse com os meus horários disponíveis.

A notícia de que seria implementada a internet no laboratório da escola possibilitou a continuidade do trabalho com o uso da plataforma on-line do professor Troy Henderson [2].

Como o laboratório possuía apenas 5 computadores adequados, foram escolhidos cinco estudantes para participarem do trabalho.

Iniciou-se a parte prática com uma aula de apresentação da plataforma on-line do professor Troy Henderson [2] e revisão sobre os comandos básicos do METAPOST relembrando os principais, conforme Fig. 5.22.

## CAPÍTULO 5. CONSTRUINDO FIGURAS COM O METAPOST



Figura 5.22: Alunos trabalhando no laboratório.

Em outra oportunidade foi ministrada aula com vários exemplos de figuras utilizadas em matemática, direcionando àquelas a serem propostas por este trabalho, sobre construção de segmentos, polígonos, ângulos, eixos, gráfico de funções, bem como o uso de rótulos contendo textos e expressões matemáticas,

Após isso os alunos ficaram livres para usar a imaginação e os conteúdos estudados para criarem figuras, conforme Figura 5.23, tirando suas dúvidas sobre utilização dos comando necessários.



Figura 5.23: Aluno desenvolvendo figura livremente.

Para finalizar, foram propostas a série de atividades apresentadas no Capítulo 5, às quais os alunos desenvolveram com boa desenvoltura e propriedade.

# Considerações Finais

O desafio deste trabalho foi utilizar uma abordagem dinâmica, apoiada no uso da tecnologia, para contribuir com um aprendizado mais efetivo e significativo em matemática.

Sendo assim, o uso do METAPOST é uma forma de ter contato, aplicar e aprofundar os diversos conceitos matemáticos estudados no Ensino Médio, principalmente em Geometria, além de oportunizar, um aprendizado na área de programação de computadores, estimulando o raciocínio lógico matemático e a organização de ideias para resolver problemas.



# Apêndice A

## Como Trabalhar com o METAPOST

A criação de gráficos com o MetaPost segue o ciclo editar-compilar-depurar conhecido de outras linguagens de programação. Para criar imagens com o MetaPost, você prepara um arquivo de texto contendo o código na linguagem do MetaPost e depois o compila. Os arquivos de entrada do MetaPost normalmente possuem a extensão `.mp`. As imagens geradas pelo METAPOST podem ser visualizadas em qualquer visualizador PostScript e também em qualquer arquivo de saída obtido através de um arquivo `TEX`.

Neste capítulo, são apresentadas as ferramentas necessárias (programas e editores para a plataforma windows) e procedimentos necessários para produzir figuras com o uso do METAPOST e inseri-la num arquivo `TEX` que por sua vez possibilita imprimir tal figura numa saída em `.pdf`, por exemplo.

### A.1 Programas necessários

Inicialmente, é necessário instalar no computador uma distribuição que implementa o TeX / LaTeX e programas relacionados. No windows, essa distribuição é o MiKTeX (pronuncia -se mick-tech ). O MiKTeX pode ser baixado no endereço <https://miktex.org/>, é gratuito e de fácil instalação.

Posteriormente, é preciso o uso de um editor de textos que possibilita desenvolver textos `TEX` e, por sua vez consegue salvar um arquivo com o formato `.mp` que é a terminologia de um arquivo METAPOST. Para tanto, podem ser usado os programas: Texmaker, Textstudio, entre outros.

Ao utilizar um destes editores e abrir um novo arquivo, deve salvá-lo com o formato

## APÊNDICE A. COMO TRABALHAR COM O METAPOST

figura-n.mp (“figura” é o nome do arquivo e “n” é um contador de figuras) na mesma pasta em que existe um arquivo  $\text{T}_{\text{E}}\text{X}$ .

### A.2 Estrutura de um documento METAPOST

A estrutura do documento METAPOST tem o seguinte formato:

```
prologues:=3;
outputformat:="mps";
outputtemplate:="%j" & "." & outputformat);
beginfig(n);
<CÓDIGO METAPOST>
endfig;
end
```

Note que o código METAPOST se encontra entre as tags `beginfig(n)`; e `endfig`; . O argumento numérico para a macro `beginfig(n)`; determina o nome do arquivo de saída correspondente, cujo nome, por padrão, é da forma `fig.n`, em que `n` é o argumento, ou seja, um contador de figuras.

Comandos também podem aparecer fora de `beginfig . . . endfig`. Essas instruções são processadas, mas as operações de desenho não geram saída visível. Normalmente, as configurações globais são colocadas fora de `beginfig . . . endfig`, por exemplo, atribuições a variáveis internas, como `outputtemplate`, `outputformat` ou um comando do preâmbulo do  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ .

Se os gráficos contiverem rótulos de texto, convém definir a variável `prologues` como 3 para garantir que as fontes corretas sejam usadas em todas as circunstâncias possíveis. A atribuição `outputtemplate` altera o esquema de nomenclatura do arquivo de saída para o formato `nomedafigura-n.mps`. Dessa forma, em vez de um índice numérico, todos os arquivos de saída obtêm uma extensão de arquivo uniforme `mps`, que é normalmente usado para a saída PostScript do METAPOST. A atribuição `outputformat` permite que o MetaPost grave arquivos de saída no formato SVG ou PNG em vez do formato PostScript.

### A.3 Inserção de figuras num documento T<sub>E</sub>X

Para inserir uma figura num documento T<sub>E</sub>X deve-se usar o ambiente `figure` que possui uma série de atributos que vão desde alinhar figuras a colocar descrição ou fazer referências a ela, a seguir temos um exemplo de como usá-lo :

```
\begin{figure}[H]
\centering
\includegraphics[scale=1]{nomedafigura-n.mps}
\caption{Aqui vai uma descrição da imagem}
\label{fig:figuran}
\end{figure}
```

Figura A.1: Inserção de figura num documento T<sub>E</sub>X

A linha `includegraphics` serve para incluir a figura especificada, no texto. Note que o arquivo referenciado para a figura possui a terminologia `.mps`, que é o arquivo executável gerado na mesma pasta do arquivo editável `.mp` quando o mesmo é compilado. Esse arquivo `.mps` deve ser copiado para a mesma pasta que contém o arquivo T<sub>E</sub>X a ser incluído a figura.

Já a linha `centering` serve para centralizar a figura. Em `caption` deve ser colocada uma descrição para a figura e `label` serve para colocar uma referência para a figura, a fim de referenciá-la em algum ponto do texto.

É necessário, para se utilizar o ambiente `figure`, colocar no preâmbulo do documento T<sub>E</sub>X a linha

```
usepackagegraphicx
```

### A.4 Visualização de figuras METAPOST em plataforma on-line

É possível visualizar figuras geradas com códigos METAPOST, de forma simples e rápida, numa plataforma on-line desenvolvida pelo professor Troy Henderson, no endereço <http://www.tlhiv.org/mppreview/>, conforme Figuras A.2 e A.3.

# APÊNDICE A. COMO TRABALHAR COM O METAPOST

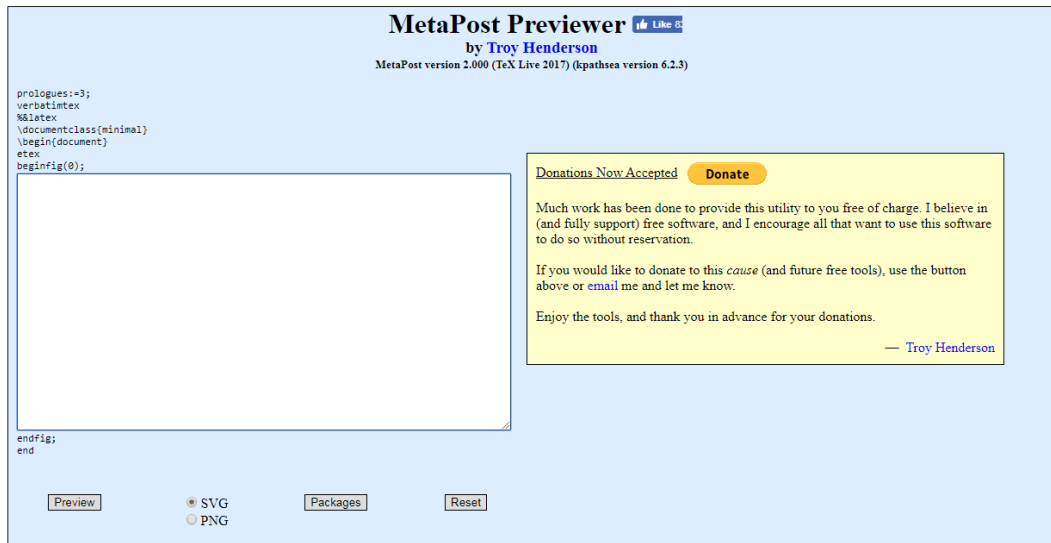


Figura A.2: Visualizador METAPOST on-line em branco

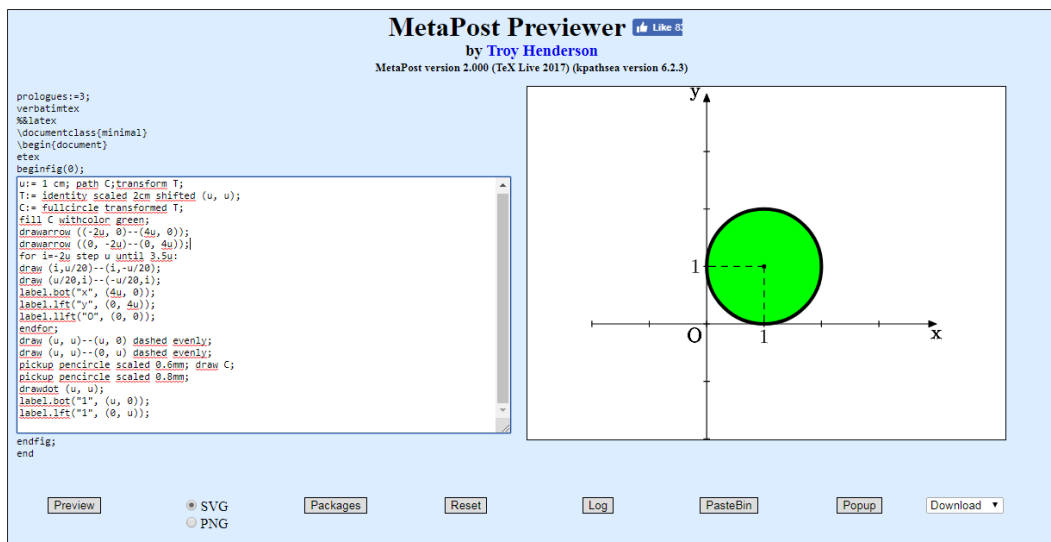


Figura A.3: Visualizador METAPOST on-line com exemplo

# Referências Bibliográficas

- [1] *DELGADO, Jorge. Geometria Analítica* - Jorge Delgado, Katia Frensel, Lhaylla Crissaff. 1.ed. Rio de Janeiro: SBM, 2013.
- [2] *HENDERSON, Troy. MetaPost Previewer*. Disponível em <http://www.tlhiv.org/mppreview/>; acesso em 02/11/2019.
- [3] *HOBBY, John D. METAPOST - A USER'S MANUAL*. Technical Report 162, AT&T Bell Laboratories, Murray Hill, New Jersey, 1992. Disponível em <http://www.tug.org/docs/metapost/mpman.pdf>; acesso em 02/11/2019.
- [4] *MUNIZ NETO, Antonio C. Fundamentos de Cálculo*. 1.ed. Rio de Janeiro: SBM, 2015.
- [5] *AUSUBEL, David P., NOVAK, Joseph D., HANESIAN, Helen. Psicologia Educacional*. Tradução Eva Nick. Rio de Janeiro: Interamericana, 1980.
- [6] *MUNIZ NETO, Antonio C. Geometria*. 1.ed. Rio de Janeiro: SBM, 2013.
- [7] *CANDIA, Estevão V. A matemática e o MetaPost*. Disponível em [http://https://sca.proformat-sbm.org.br/sca\\_v2/get\\_tcc3.php?id=160350039/](http://https://sca.proformat-sbm.org.br/sca_v2/get_tcc3.php?id=160350039/); acesso em 02/11/2019.
- [8] *VICENTE, Lucas. Escrevendo Imagens com MetaPost*. Disponível em <http://www.ime.usp.br/~alkaid/metapost.pdf>; acesso em 02/11/2019.
- [9] *LIMA, Elon Lages Coordenadas no Plano*. 5.ed. Rio de Janeiro: SBM, 2011.
- [10] *HENDERSON, Troy; HENNIG, Stephan. A Beginner's Guide to METAPOST for Creating High-Quality Graphics*. 2013. Disponível em <http://www.tug.org/docs/metapost/mpintro.pdf>; acesso em 02/11/2019.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [11] *HECK, André. Tutorial in MetaPost.* AMSTEL Institute, 2003, disponível em <http://tex.loria.fr/prod-graph/heck-metapost2003.pdf>; acesso em 02/11/2019.
- [12] **MetaPost: exemples.** A (La)Tex Encyclopaedia, disponível em <http://tex.loria.fr/prod-graph/zoonekynd/metapost/metapost.html>; acesso em 02/11/2019.
- [13] *HOBBY, John D. Drawing Graphs with METAPOST.* Technical Report 164, AT&T Bell Laboratories, Murray Hill, New Jersey, 1992. Disponível em <https://www.tug.org/docs/metapost/mpgraph.pdf>; acesso em 02/11/2019.
- [14] *DEPARTAMENTO DE EDIÇÕES EDUCATIVAS DA EDITORA MODERNA Matemática, construção e significado.* 1.ed. São Paulo: Editora Moderna, 2005.
- [15] *DANTE, Luiz Roberto. Matemática: Contexto e Aplicações - 3.ed.* São Paulo: Ática, 2008.