MATEMÁTICA PARA A COMPUTAÇÃO

FABRÍCIO GALENDE MARQUES DE CARVALHO

AVISO SOBRE DIREITOS AUTORAIS

Sumário

PREFÁCIO

Esse material tem como objetivo fornecer ao estudante de cursos de computação uma visão geral de alguns tópicos da matemática superior com enfoque na computação.

O prefixo **EP** é usado para identificar **Exercícios e Problemas** que têm como objetivo fazer com que o estudante retenha termos, conceitos e definições estudados na unidade, além de aplicá-los à resolução de problemas hipotéticos e teóricos "no papel". Exercícios com prefixo **PC** são exercícios de **Prática Computacional** e tem como foco a aplicação dos tópicos estudados à resolução de problemas através da utilização de ferramentas computacionais, tais como simuladores, linguagens de programação, etc. Ao final de cada unidade são apresentados exercícios resolvidos, prefixo **ER**, que exemplificam a utilização de conceitos e técnicas estudados na seção imediatamente anterior.

1. INTRODUÇÃO

1.1. MOTIVAÇÃO PARA O ESTUDO DA MATEMÁTICA EM CURSOS DE COMPUTAÇÃO

Originalmente, computadores foram concebidos para efetuar cálculos. As aplicações que foram inicialmente desenvolvidas eram voltadas às finalidades científicas ou bélicas.

Matemáticos, engenheiros, físicos e outros cientistas de áreas correlatas tiveram papel crucial na criação dos dispositivos computacionais e, também, na concepção e análise de toda uma teoria por trás da computação. É natural, portanto, que a computação tenha permeado tais áreas do conhecimento.

Apesar do fato de computadores terem se tornado dispositivos capazes de executar aplicações com múltiplos propósitos, desde aplicativos de entrega de comida até mesmo sistemas de guiamento e controle de satélites, em sua essência, computadores continuam a operar sobre números e todas as aplicações existentes são baseadas em princípios bem fundamentados da matemática. Dessa forma, é fundamental que um desenvolvedor de sistemas computacionais, seja ele de software ou hardware, compreenda os fundamentos matemáticos vinculados à computação e que seja capaz de aplicá-los à concepção, construção, avaliação e análise de tais sistemas.

1.2. DIFERENÇA ENTRE MATEMÁTICA ELEMENTAR E MATEMÁTICA DO ENSINO SUPERIOR

Tipicamente, a matemática ensinada ao longo do ensino fundamental e ensino médio tem um enfoque em aspectos básicos, tais como operações aritméticas, álgebra elementar e geometria. As aplicações estudadas por vezes são simplificações extremas da realidade e, com frequência, não refletem a natureza de vários fenômenos reais. Além disso, essa matemática elementar possui um enfoque pouco analítico e muito mais intuitivo e computacional (i.e., no sentido de execução mecânica de cálculos).

A matemática do ensino superior, apesar de ser construída sobre os pilares da matemática elementar, tem um enfoque muito mais analítico, por vezes extremamente formal, e procura fornecer ao estudante ferramentas que sejam capazes de resolver problemas reais, cujas simplificações não comprometem a eficácia da solução obtida para um dado problema.

Em geral, os aspectos formais ficam evidenciados quando **definições precisas** são fornecidas e resultados são demonstrados através de **provas matemáticas.**

Para o caso de aplicações, em geral são construídos e estudados **modelos matemáticos** que devem ser próximos o suficiente da realidade de modo que forneçam dados e informações úteis à solução de problemas. Para o caso de aplicações computacionais, há modelos que se relacionam a áreas da computação tais como banco de dados, aprendizagem de máquina, simulação de fenômenos, computação gráfica, economia e finanças, etc.

1.3. ALGUMAS ÁREAS DE ESTUDO DA MATEMÁTICA E SUAS APLICAÇÕES À COMPUTAÇÃO

Apesar da matemática ser uma única ciência, é comum que esta seja dividida em diferentes disciplinas (ou áreas) para facilitar a definição do escopo de estudo e, também, auxiliar o estudante no processo de aprendizado e, consequentemente, na adequada seleção do ferramental necessário à resolução de um dado problema.

A seguir, são enumeradas algumas das áreas frequentemente estudadas pela matemática superior, com um enfoque em computação:

- **ARITMÉTICA:** Estuda as operações fundamentais existentes em um sistema numérico. As regras de precedência, agrupamento e de obtenção de resultados de operações numéricas estão entre as aplicações mais comuns na área de computação.
- **ÁLGEBRA ELEMENTAR:** Estuda as propriedades e operações envolvendo símbolos que representam objetos “arbitrários” denominados de variáveis. Equações matemáticas, polinômios, etc. , estão dentro do objeto de estudo da álgebra elementar. Expressões algébricas com frequência são utilizadas em modelos computacionais de sistemas.
- **ALGEBRA ABSTRATA:** Trata do estudo de estruturas matemáticas tais como anéis, grupos, corpos, etc. Estas estruturas são abstratas no sentido de não necessitarem de uma correspondência direta ou intuitiva com algo real, tangível. Geralmente o estudo da álgebra abstrata é conduzido nos cursos de bacharelado em matemática ou cursos de pós-graduação em ciências exatas tais como computação. Trata-se de um ramo da matemática formal e não computacional (i.e., não mecânica), mas cujos resultados abstratos e gerais são usados para fundamentar resultados específicos e computacionais (ou seja, uma demonstração abstrata e geral pode ser utilizada para se chegar a um resultado tangível particular, dado que as condições utilizadas na demonstração abstrata sejam satisfeitas pelo ente real.
- **ÁLGEBRA LINEAR:** Pode ser considerada como uma área específica da álgebra que estuda expressões, equações e sistemas de equações algébricas que possuem uma

forma específica com relação às variáveis (e.g: uma equação do tipo $y = 10x$ é uma equação linear). A álgebra linear é uma das áreas da matemática que possui vasta aplicação em computação, podendo ser destacadas as áreas de aprendizagem de máquina, computação gráfica, entre outras.

- **ALGORITMOS:** Constituem a área da matemática que se preocupa com os métodos sistemáticos de computação para a obtenção de soluções aos problemas. Incluem a análise de existência de soluções expressas através de algoritmos (computabilidade), análise de desempenho, estruturas de dados, etc.
- **LÓGICA MATEMÁTICA:** É a área da matemática que está voltada à sistematização do processo de raciocínio na obtenção de conclusões. A lógica matemática lida com o processo de inferência, ou seja, a conclusão a partir de um certo conjunto de informações ou fatos (denominados de premissas). Os sistemas computacionais especialistas (e.g.: sistemas de diagnóstico) são fundamentados na lógica matemática.
- **MATEMÁTICA DISCRETA**: A noção de discreto tem a ver com separação clara e com contagem. A matemática discreta lida com estruturas/objetos que podem ser facilmente separados uns dos outros e facilmente contados. Um exemplo de objetos que podem ser facilmente contados são número de celulares armazenados em uma agenda, já a quantidade de água em uma garrafa, fica difícil de ser separada e contabilizada (e.g: 1L, 1.1L, 1.1000001L, etc.). Esse ramo da matemática é um dos pilares da computação, dado que computadores digitais modernos operam utilizando dados que são iminentemente discretos. Diferentemente de áreas como o cálculo e a análise matemática, a matemática discreta faz uso frequente de métodos numéricos (aproximados) para resolver problemas do mundo real que envolvem objetos contínuos.
- **TEORIA DE CONJUNTOS:** Estuda o agrupamento de itens (que podem ou não ter alguma relação entre si) e, também, a associação entre esse agrupamento a outros (operações entre conjuntos). A teoria de conjuntos e uma das bases dos sistemas de bancos de dados.
- **RELAÇÕES E FUNÇÕES:** É a área da matemática que está voltada a associação entre elementos que pertencem a dois ou mais conjuntos. Relações e funções são utilizados como base da modelagem e construção de sistemas computacionais (e.g.: definição de funções em uma linguagem de programação). Em geral, ao se modelar um fenômeno do mundo real, busca-se a construção de um modelo que reflita uma dependência funcional entre duas ou mais variáveis.
- **GEOMETRIA:** É a área da matemática que estuda elementos tais como formas, tamanhos e posição. Considera aspectos espaciais. Uma das especializações da geometria é a trigonometria, que estuda relações entre ângulos e dimensões considerando como base a figura geométrica de um triângulo. A geometria é uma das áreas que está presente em praticamente todas as aplicações computacionais modernas, pois as interfaces gráficas são construídas levando-se em conta aspetos geométricos.
- **CÁLCULO E ANÁLISE MATEMÁTICA:** O cálculo e a análise matemática estudam as funções sob o ponto de vista de variação, considerando elementos infinitamente pequenos. Enquanto o cálculo possui uma natureza mais prática e operacional, a análise efetua esse estudo de maneira formal. Essas áreas encontram aplicabilidade direta na simulação computacional incluindo o projeto e análise de sistemas aeroespaciais, na resolução de problemas de otimização, entre outros. Os modelos matemáticos que dependem de aspectos do cálculo tais como derivadas e integrais, são ditos modelos de sistemas dinâmicos.

- **PROBABILIDADE:** Estuda o quão factível um evento é ou não. Está atrelada ao estudo não só da aleatoriedade, mas também da regularidade presente em tais eventos. A teoria da probabilidade encontra vasta aplicação na área de simulação computacional, desenvolvimento de jogos, processamento de linguagem natural, entre outras.
- **ESTATÍSTICA:** A estatística lida com o agrupamento, organização, exibição, análise e interpretação de dados. A estatística é amplamente utilizada no processo de tomada de decisão automatizada por parte de sistemas computacionais "inteligentes". Além disso, a estatística é um dos principais pilares da análise de dados em sistemas computacionais.

EXERCÍCIO RESOLVIDO

ER.1.1. Identifique áreas da matemática que são aplicáveis à resolução do seguinte problema:

"O usuário de um sistema computacional abre uma imagem contendo sua foto e percebe que ao tirar sua selfie seu corpo ficou ligeiramente inclinado. Aplicando uma função de rotação do software editor de imagens, aplica uma rotação no sentido anti-horário, no valor de 15º e percebe que a imagem agora está corrigida."

Resolução: Trata-se de um problema envolvendo **relações espaciais** entre os elementos de uma imagem. Esse problema é abordado na geometria e, também, na **álgebra linear** para o caso de rotações simples em imagens.

EXERCÍCIOS

EP. 1.1. Cite uma referência (livro, artigo de internet, vídeo, etc.), para cada uma das áreas de estudo da matemática citadas a seguir:

a) Aritmética

b) Álgebra elementar

c) Álgebra linear

d) Algoritmos

f) Lógica matemática

g) Teoria dos conjuntos

h) Relações e funções

i) Geometria

j) Cálculo e análise matemática

k) Probabilidade

l) Estatística

EP.1.2. Ilustre uma fórmula matemática/função, ou equação, ou diagrama, ou símbolo, representativo das seguintes áreas da matemática:

a) Álgebra linear

b) Teoria de conjuntos

c) Probabilidade

d) Cálculo / análise matemática

e) Geometria (trigonometria)

f) Lógica matemática

EP. 1.3. Identifique uma área de conhecimento da matemática que pode ser utilizada para a resolução dos seguintes problemas (obs: caso envolva múltiplas áreas, cite-as e diga qual é a que aparenta ser predominantemente utilizada. Em alguns, casos mais de uma combinação de áreas é possível).

a) Determinação de quais palavras são mais utilizadas em uma língua, tal como o português ou inglês.

b) Desenvolvimento de um sistema de recomendação de produtos considerando o histórico dos produtos que são frequentemente comprados em um site de vendas. Esse sistema requer a organização, a análise e a adequada exibição desses dados.

c) Cálculo de trajetórias de lançamentos oblíquos de objetos sujeitos a atração gravitacional. Nesse caso, os objetos estão sujeitos a aspectos dinâmicos (velocidades, acelerações, etc.).

d) Determinação de cantos e bordas em imagens digitais, considerando a variação que ocorre na intensidade dos pixels da imagem. Nesse caso, considera-se que os pixels são estruturas discretas (ou seja, pertencentes a uma grade NxM) e que os valores das intensidades também são discretos.

e) Definição de um conjunto de passos computacionais a serem executados de modo a processar um conjunto de entradas e obter uma saída específica. Análise desse conjunto de passos quanto a sua correção, consumo de memória, tempo, etc.

f) Desenvolvimento de um jogo envolvendo objetos que se movem de modo “aleatório” em uma interface gráfica e que devem ser destruídos por um canhão direcionado através dos comandos de teclado fornecidos pelo usuário.

g) Um assistente pessoal digital, acionado através de comandos de voz. Os comandos de voz são processados de acordo com a análise e processamento de um histórico de dados utilizado para o treinamento do modelo (quanto mais dados forem utilizados no treinamento, mais acurado será o assistente pessoal digital).

EP.1.4. Uma prova matemática é um raciocínio dedutivo que faz uso de informações disponíveis para chegar a uma conclusão. Na matemática, em áreas como aritmética e álgebra, são comuns provas que fazem uso de propriedades simples tais como comutatividade, distributividade, utilização de elemento neutro na soma ou multiplicação (i.e., somar ou multiplicar ambos os lados de uma equação por um mesmo valor). Considere a seguinte prova matemática:

Provar que (-1) x (-1) = + 1

Prova:

1 -1 = 0	*Um número somado ao seu oposto é igual a zero.*
(-1) x (1-1) = (-1) x 0	*Multiplicar ambos os lados de uma equação por um número diferente de zero não altera a equação.*
(-1)x1 + (-1)x(-1) = 0	*Distributividade.*
-1 + (-1)x(-1) + 1= +1	*Somar um mesmo valor a ambos os lados de uma equação não altera a equação.*
-1 + (-1)x(-1) + 1= +1	*Um número somado ao seu oposto é igual a zero.*
(-1) x (-1) = +1	*Cqd (como queríamos demonstrar).*

Considerando o resultado anterior, escreva uma prova para a seguinte proposição:

Proposição:

(-10) * (-3) = + 30

2. DEFINIÇÕES MATEMÁTICAS

Definições matemáticas são maneiras de se esclarecer o significado de objetos e símbolos utilizados em matemática. Evitam que equívocos sejam cometidos ao se ler ou utilizar um determinado objeto ou símbolo.

Em computação, definições são amplamente utilizadas para criação de **variáveis** que serão vinculadas a valores, criação de modelos representativos de objetos do mundo real (e.g.: classes), criação de símbolos fáceis de serem memorizados para representar conjuntos de valores menos intuitivos numericamente (e.g.: MES.JANEIRO correspondente ao valor 1), etc.

Uma definição é composta por um definiendum e por um definiens. Matematicamente é comum que seja representada tal como:

$$definiendum \equiv definiens$$

$$definiendum \triangleq definiens$$

$$definiendum \stackrel{\text{def}}{=} definiens$$

Definições podem ser denotativas ou conotativas.

2.1. DEFINIÇÕES DENOTATIVAS

São definições que especificam o conjunto (agrupamento) de objetos que correspondem ao *definiendum*.

Alguns tipos de definições denotativas são através de enumeração, definição ostensiva e definição recursiva.

2.1.1. ENUMERAÇÃO

É feita através da exemplificação dos objetos correspondentes ao *definiens*.

Exemplo:

Número par $\stackrel{\text{def}}{=}$ 2, 4, 6, 8, 10, ...

Em computação, definições desse tipo são comuns em tipos de dados denominados de **enumerações.** Enumerações são utilizadas para tornar programas de computadores mais legíveis, evitando equívocos e facilitando o uso por parte do programador.

Em algoritmos e programas, enumerações são tipicamente definidas da seguinte forma:

```
Enumeração Nome_da_enumeração:
     símbolo 1 = valor para o símbolo 1
     símbolo 2 = valor para o símbolo 2
     ...
     símbolo N = valor para o símbolo N
Fim-enumeração
```

2.1.2. DEFINIÇÃO OSTENSIVA

Também conhecida como definição "apontando para o objeto". Tipicamente é utilizada quando o *definiens* é difícil ou muito complicado de ser esclarecido através de palavras, símbolos, etc.

Exemplo:

Uma caneta esferográfica $\stackrel{\text{def}}{=}$ 

Nesse caso, imagine que aquele que efetua a definição observa e aponta para uma caneta enquanto que outra pessoa olha para a caneta e associa isso ao nome "caneta esferográfica).

2.1.3. DEFINIÇÃO RECURSIVA

São definições que se baseiam em uma definição simples inicial, denominado de caso base, efetuam uma generalização, denominada de cláusula indutiva, e, também, possuem uma cláusula que garante que objetos definidos de modo recursivos segundo as cláusulas anteriores sempre satisfarão a definição (cláusula externa ou de fechamento).

Exemplo:

Menor inteiro positivo $\stackrel{\text{def}}{=}$ 1 (caso base).

Se N é um número inteiro positivo, então N+1 também é (cláusula indutiva).

Número inteiro positivo $\stackrel{\text{def}}{=}$ Número que satisfaz as duas condições anteriores.

Definições recursivas são muito utilizadas em computação na criação de funções e estruturas de dados. Ao se definir uma estrutura ou função de modo recursivo, sempre é necessária a criação do caso base que, nesse caso, é o responsável por evitar que o algoritmo nunca termine. A ausência de término, nesse caso, acontece justamente por causa da cláusula indutiva.

Em **expressões matemáticas**, mais especificamente **fórmulas**, definições recursivas costumam utilizar subscritos para relacionar **variáveis** (i.e.: símbolos que representam números não explicitamente especificados) cujos valores dependem dessas mesmas variáveis mas em situações ou instantes distintos. Isso corresponde a cláusula indutiva ou relação de recorrência.

Exemplo:

Primeiro termo da sequência numérica $\stackrel{\text{def}}{=}$1

Termos da sequência numérica $\stackrel{\text{def}}{=} x_k$, com x_k expresso pela fórmula $x_k = 2.\ x_{k-1}$

Essa definição é equivalente a

Sequência numérica $\stackrel{\text{def}}{=}$ {1, 2, 4, 8, ...}

Que também pode ser expressa segundo a seguinte notação matemática para conjuntos:

Sequência numérica = $\{x_k \mid x_1 = 1 \text{ e } x_k = 2x_{k-1} \text{ para } k \geq 2\}$

2.2. DEFINIÇÕES CONOTATIVAS

São definições que especificam critérios que devem ser satisfeitos ou o intuito de determinado objeto para que possa corresponder a um dado *definiendum.*

2.2.1. DEFINIÇÃO POR SINÔNIMO

São definições que proveem um termo ou símbolo com um mesmo significado daquilo que está sendo definido.

Exemplo:

Enorme $\stackrel{\text{def}}{=}$ grande, extenso

2.2.2. DEFINIÇÃO OPERACIONAL

Utiliza atributos que podem ser observados e que estão atrelados a como o *definiendum* é utilizado em alguma operação ou procedimento. Tipicamente especifica algo que é empiricamente testável.

Exemplo:

Símbolo de adição $\stackrel{\text{def}}{=}$ é o símbolo utilizado em expressões de soma de dois números, tais como em $8 + 10$.

Ferro = metal que produz óxido/ferrugem ao ser deixado em contato com a água e com oxigênio.

2.2.3. DEFINIÇÃO POR GÊNERO E DIFERENÇA

Efetuada identificando-se uma classe ou categoria a qual o objeto pertence e depois refinando-se para suas características mais específicas dentro dessa categoria.

Exemplo:

Ser humano $\stackrel{\text{def}}{=}$ animal mamífero e racional

Na definição anterior, animal é a categoria mais geral enquanto que mamífero e racional são refinamentos ou especializações.

Em computação definições por gênero e diferenças são comuns em linguagens de programação com suporte a orientação a objetos quando se efetuam definições através de **herança**.

EXERCÍCIO RESOLVIDO

ER.2.1. Defina, utilizando enumeração, os números naturais ímpares. Nesse caso é possível enumerar todos esses números?

Resolução:

Números ímpares $\stackrel{\text{def}}{=}$ 1, 3, 5, 7, ...

Note-se que se trata de um conjunto enumerável mas cuja quantidade de elementos é infinita. Portanto, não é possível enumerar todos os elementos do conjunto, sendo utilizado nesse caso o símbolo ... (reticências, que significam que a sequência numérica continua indefinidamente seguindo o padrão já ilustrado, ou seja, cada elemento obtido somando-se 2 ao anterior.)

ER.2.2. Defina, utilizando recursão, os membros de uma família de pessoas que começou com um homem e uma mulher denominados de Adão e Eva.

Resolução:

Primeiros membros da família $\stackrel{\text{def}}{=}$ Adão e Eva (1)

Se uma pessoa é da família de Adão e Eva (geração k), então os filhos dessa pessoa (geração k+1) também são da família de Adão e Eva. (2)

Para ser da família de Adão e Eva, então as condições (1) e (2) devem ser satisfeitas simultaneamente.

EXERCÍCIOS

EP.2.1. Efetue uma definição, através de enumeração, de todos os membros da sua família, considerando apenas parentesco direto de pais, irmãos e filhos.

EP.2.2. Efetue uma definição, através de enumeração, de cursos de graduação existentes na instituição de ensino superior em que você está estudando.

EP.2.3. Efetue uma definição, através de enumeração, de números primos inferiores a 40. Um número é dito primo se só possui como divisor ele mesmo e o número um.

EP.2.4. Efetue uma definição ostensiva, de um colega de classe.

EP.2.5. Efetue uma definição recursiva de uma sequência numérica cujos elementos são obtidos multiplicando-se, a partir do segundo elemento, o elemento anterior por 3. Considere que o primeiro elemento vale 2.

PC.2.1. Utilizando linguagem de programação defina:

a) Uma enumeração para os meses do ano.

b) Uma enumeração para os dias da semana.

c) Uma função recursiva para o cálculo do fatorial de um número.

d) Uma definição que corresponda a definição do tipo gênero-diferença para um uma pessoa que estude em uma faculdade. Utilize uma linguagem que dê suporte a herança. Suponha que toda pessoa possua nome e idade e que um estudante de uma faculdade possui registro acadêmico e ano e semestre de ingresso.

PC.2.2. Um veículo possui a capacidade de se mover, expressa pela alteração na sua coordenada de longitude e latitude. Um veículo elétrico é um veículo que possui como fonte de energia primária a eletricidade (armazenada em uma bateria, que tem sua capacidade especificada em

Ampere-hora). Um veículo elétrico e voador é um veículo que também possui a capacidade de se mover na vertical, expressa pela alteração de sua altitude em relação ao solo. Represente um veículo elétrico e voador utilizando uma cadeia de herança. Defina o código-fonte representativo do modelo em um arquivo separado daquele que faz uso desse e, adicionalmente exemplifique o acesso e a modificação desses atributos através de chamada de suas operações (moverHorizontal, moverVertical).

PC.2.3. O que acontece ao se executar uma chamada a uma função recursiva que chama a si mesma um elevado número de vezes? Dê um exemplo utilizando o código-fonte da progressão aritmética fornecido pelo professor. Faça um comparativo escrevendo um algoritmo e código que sejam equivalentes ao recursivo em termos de entradas e saídas mas que utilizem iteração ao invés de recursão. Qual sua conclusão?

PC.2.4. Uma progressão geométrica é uma sequência numérica onde cada elemento, a partir do segundo, é obtido multiplicando-se o anterior por uma constante. Utilizando uma linguagem de programação que dê suporte a orientação a objetos, defina uma progressão geométrica e dê exemplo de geração de seus primeiros 50 termos.

PC.2.5. A sequência de Fibonacci é definida da seguinte forma: {1, 1, 2, 3, 5, 8, 13, ...}, ou seja, para $k>2$, $x_k = x_{k-1} + x_{k-2}$. Utilizando uma linguagem de programação com suporte a orientação a objetos, defina uma classe que modele a sequência de Fibonacci e exemplifique o cálculo de alguns de seus termos. Ilustre a chamada recursiva e identifique chamadas repetidas a um mesmo valor construindo uma árvore de chamadas.

PC.2.6. Modele, utilizando orientação a objetos, uma classe que representa uma sequência numérica onde os termos, a partir do 4º são obtidos somando-se os três termos anteriores. Considere que os três termos iniciais da sequência são parâmetros do construtor do objeto representativo da sequência. Utilize recursão e demonstre a exibição dos 15 primeiros termos da sequência, para diferentes valores de a_1, a_2 e a_3.

3. TEORIA DE CONJUNTOS

3.1. IDEIA GERAL DE CONJUNTO

Em geral conjunto remete a uma coleção ou classe de elementos. Nessa coleção, não há restrição referente ao tipo de elementos (e.g.: letras, números, palavras, etc.).

Em geral, na matemática, conjuntos são representados por letras maiúsculas e seus elementos por letras minúsculas.

Em um conjunto, não são consideradas nem a ordem nem a repetição de seus elementos.

Exemplos:

Conjunto A = {a, b} = {b, a}

Conjunto B = { x, y, z}

Conjunto C ={"carro", "moto", 10}

Quando um elemento *e* faz parte de um conjunto *E*, diz-se que o elemento pertence ao conjunto e representa-se da seguinte forma:

$$e \in E.$$

Lê-se "o elemento e pertence ao conjunto E", caso contrário diz-se que o elemento não pertence ao conjunto e representa-se por

$$e \notin E.$$

3.2. NOTAÇÃO E DESCRIÇÃO

Além da representação usual por letras maiúsculas, conjuntos também podem ser representados de modo mais detalhado através da enumeração e seus elementos (***roster notation***) ou através da descrição de uma propriedade característica desses mesmos elementos (***set builder notation***).

Exemplos:

Roster notation:

A ={ 2, 4, 6, 8, 10}

B = {a, b, c, d, e, f, g, ..., z}

Set builder notation:

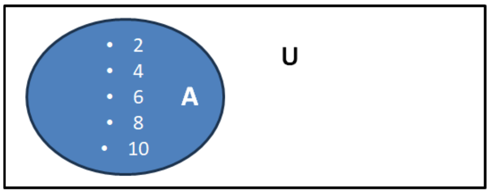
C = { x | x é letra do alfabeto português}

D = { y | (y = 2x -1) e (x > 1) e (x é número natural) }

D = { y | (y = 2x -1) $\wedge$ (x > 1) $\wedge$ (x é número natural) }

Observação: Na notação *set builder*, os conectivos *e* e *ou* são representados por $\wedge$ e $\vee$, respectivamente. Um $\wedge$ significa que ambas as condições (ou restrições) devem ser satisfeitas, já um $\vee$ significa que pelo menos uma das condições deve ser satisfeita, não requerendo que todas sejam satisfeitas simultaneamente.

Em termos gráficos, é comum que conjuntos sejam representados utilizando-se o diagrama de Veen.



Ao se utilizar a notação de Venn, é comum, também que o conjunto que contém todos os valores possíveis para um determinado conjunto também seja representado. Nesse caso, trata-se do conjunto universo (*U*), representado por um retângulo que contém internamente uma elipse representativa do conjunto específico. No caso do exemplo anterior, o conjunto *A*.

Alguns conjuntos numéricos possuem letras/símbolos especiais, são eles:

- ✓ Conjunto dos números **naturais**: $\mathbb{N}$
- ✓ Conjunto dos números **inteiros**: $\mathbb{Z}$
- ✓ Conjunto dos números **racionais**: $\mathbb{Q}$
- ✓ Conjunto dos números **irracionais**: ($\mathbb{R}$ - $\mathbb{Q}$)
- ✓ Conjunto dos números **reais**: $\mathbb{R}$
- ✓ Conjunto dos números **complexos**: $\mathbb{C}$

3.3. CASOS ESPECIAIS DE CONJUNTOS NUMÉRICOS

3.3.1. CONJUNTO DOS NÚMEROS NATURAIS

É formado pelos números utilizados em contagens simples.

$$\mathbb{N}=\{1, 2, 3, ...\}$$

Em algumas referências é comum que seja incluído o 0 como parte desse conjunto.

Possui propriedade de fechamento (*closure*) para soma e multiplicação, ou seja, a soma e a multiplicação de dois números naturais sempre resultam em um número natural.

!

- ✓ Em linguagens de programação, números naturais são frequentemente utilizados para indexar agrupamentos de objetos, tais como listas, vetores (arrays) e matrizes (arrays multidimensionais).
- ✓ Quando o índice utilizado para indexar coleções resulta de uma operação distinta de soma ou multiplicação envolvendo dois números naturais, não é garantido que o índice resultante será um número natural, portanto, pode ocorrer erro de indexação. Nesse caso, algumas linguagens de programação retornam valores indefinidos ou então há o lançamento de uma exceção.

3.3.2. CONJUNTO DOS NÚMEROS INTEIROS

Trata-se de uma extensão dos números naturais que inclui números negativos e o zero.

$\mathbb{Z} = \{..., -3, -2, -1, 0, 1, 2, 3, ... \}$

Além da adição e multiplicação, possui também fechamento para a subtração.

Em vários algoritmos, os números inteiros são utilizados para representar a indexação em sentido oposto ao usual em estruturas como vetores. Nesse caso, há a omissão do detalhe de indexação de modo que o usuário "percebe" um número negativo como um acesso no sentido do fim para o início.

Exemplo:

Considerando um vetor `x = [1, 2, 3, 4, 5]`, o elemento `x.get(-1)` pode significar o primeiro elemento começando-se do fim para o início. Ou seja, na verdade a indexação é feita da seguinte forma:

```
Classe array{
      valores: vetor [0..N-1] de números

      Função get(indice): número
            Se indice >=0 então
                  retorne valores[indice]
            Senão
                  indice ← N-indice
                  retorne valores[indice]
            Fim-se
      Fim-função
```

3.3.3. CONJUNTO DOS NÚMEROS RACIONAIS

Trata-se do conjunto formado pelos números inteiros e pelos números fracionários que podem ser escritos sob a forma p/q.

$$\mathbb{Q} = \{ p/q \mid p,q \in \mathbb{Z} \wedge q \neq 0 \}$$

Possui fechamento para a soma, subtração, multiplicação e divisão.

Inclui os números fracionários com as representações finitas e infinitas periódicas.

Os números pertencentes ao conjunto dos números racionais são amplamente utilizados nas mais diversas áreas de aplicação, incluindo finanças, engenharia, computação gráfica, etc.

Como os computadores são capazes somente de representar números utilizando um número finito de casas decimais, pode-se afirmar que os números racionais são utilizados em praticamente todas as aplicações computacionais. Quando um número não é racional, é necessário convertê-lo, por aproximação, para um número racional representável.

3.3.4. CONJUNTO DOS NÚMEROS IRRACIONAIS

É formado pelos números que não podem ser postos na forma de um número racional.

Esses números podem resultar da razão de dois números não comensuráveis (i.e., que não podem ser postos em uma referência comum de comparação), raízes de números primos, etc.

$$\mathbb{R} - \mathbb{Q} = \{ \pi, -\pi, \sqrt{2}, \sqrt{3}, \ldots \}$$

!

- ✓ Em várias linguagens de programação, há bibliotecas numéricas que possuem constantes contendo aproximações para os números racionais com um número razoável de casas decimais. Sempre que possível, é recomendável utilizar essas constantes aproximadas no desenvolvimento de aplicações.
- ✓ Caso a linguagem de programação não possua tal recurso, em sistemas de alto desempenho, é recomendável que o desenvolvedor efetue o cálculo prévio, armazene em arquivo e carregue os valores aproximados nas aplicações, objetivando assim evitar o *overhead* de cálculo e aproximação.

3.3.5. CONJUNTO DOS NÚMEROS REAIS

É formado pela intersecção entre o conjunto dos números racionais e irracionais.

Esse conjunto é representado pela letra $\mathbb{R}$.

Em computação, os números reais possuem sempre representação fracionária finita e são conhecidos como números de ponto flutuante (*float point number*).

3.3.6. CONJUNTO DOS NÚMEROS COMPLEXOS

Formado por números com o formato *(a+bi)*, sendo que a e b são números reais, chamados de parte real e imaginária, respectivamente.

$$\mathbb{C} = \{a + bi \mid (a,b \in \mathbb{R}) \wedge (i = \sqrt{-1}\ )\}$$

Números complexos são utilizados em sistemas computacionais de engenharia, física, etc.

Um número real pode ser visualizado em um sistema de coordenadas cartesianas onde o número *b* é representado no eixo das ordenadas *(y)* e o número a no eixo das abscissas *(x).*

Exemplo:

Número imaginário *n = 3 + 8i*

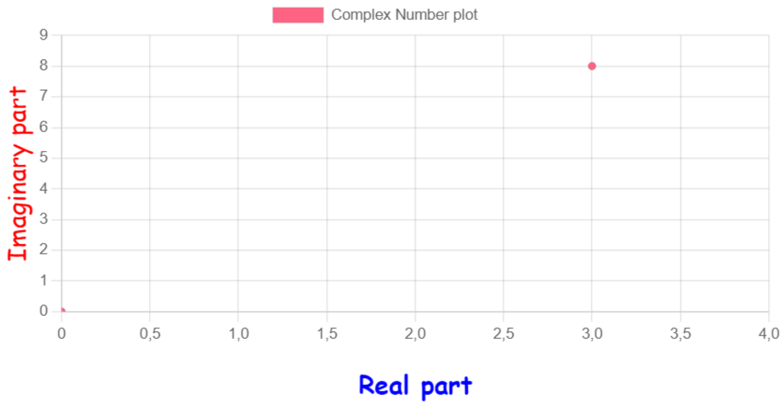


Figura 1. Representação de um número imaginário no plano cartesiano.

Um outro exemplo de aplicação dos números complexos é a geração de imagens com padrões que se repetem e que são interessantes. Algumas dessas imagens são denominadas de fractais.

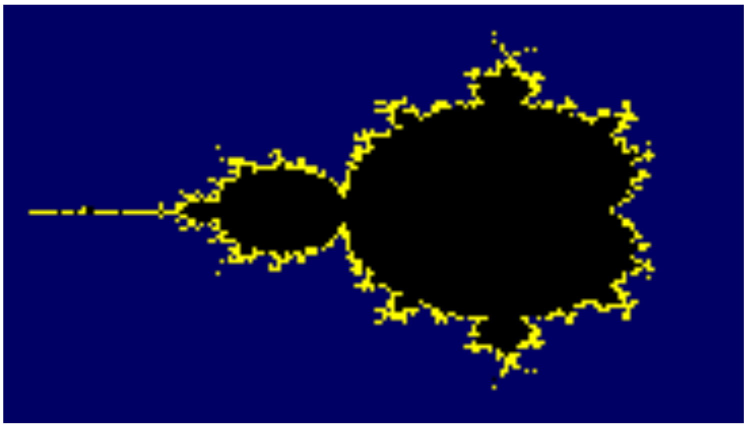


Figura 2. Conjunto de Mandelbrot.

3.4. SÍMBOLOS ESPECIAIS PARA ALGUNS CONJUNTOS

Seja X um conjunto numérico arbitrário, então:

X^* é o mesmo conjunto excluindo-se o 0.

X_+ é o mesmo conjunto excluindo-se os elementos negativos.

X_- é o mesmo conjunto excluindo-se os elementos positivos.

Um conjunto que também tem uma representação especial é o conjunto sem elementos, denominado de conjunto vazio:

$$\{\,\} \text{ ou } \varnothing$$

3.5. CONJUNTOS FINITOS E CARDINALIDADE

É um conjunto que contém um número finito de elementos. Nesse caso, o número de elementos do conjunto é denominado de cardinalidade do conjunto, que, para um conjunto *E* é representado por:

$$| E |$$

3.6. RELAÇÕES E OPERAÇÕES ENTRE CONJUNTOS

3.6.1. IGUALDADE ENTRE CONJUNTOS

Dois conjuntos A e B são ditos iguais se e somente se possuem os mesmos elementos.

$$A = B \leftrightarrow x\,(x \in A \wedge x \in B)$$

Se existir ao menos um elemento que não faça parte de ambos os conjuntos, então esses conjuntos são diferentes ($A \neq B$)

3.6.2. INCLUSÃO/SUBCONJUNTOS

Um conjunto *A* é subconjunto (***subset***) de B quando todo elemento de *A* é também elemento de B.

$$A \subseteq B \leftrightarrow \forall x\,(x \in A \rightarrow x \in B)$$

Quando A é diferente de B, diz-se que A é um subconjunto próprio de B (***proper subset***).

$$A \subset B \leftrightarrow \forall x\,(x \in A \rightarrow x \in B) \wedge (A \neq B)$$

3.6.3. CONJUNTO DAS PARTES DE UM CONJUNTO

Também conhecido como Power Set, é o conjunto que é construído a partir de todos os subconjuntos de um conjunto *A*:

$$P(A) = \{X \mid X \subseteq A\}$$

Exemplo: Determinar o *power set* do conjunto A = {1, 2}

$$P(A) = \{\varnothing, \{1\}, \{2\}, \{1,2\}\}$$

3.6.4. UNIÃO ENTRE CONJUNTOS

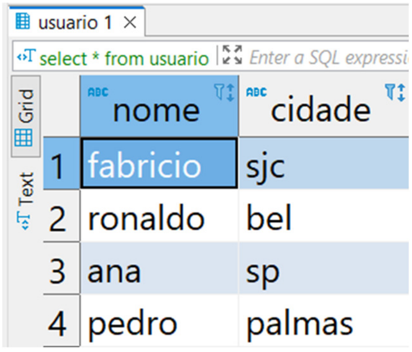
Considerando dois conjuntos *A* e *B*, a união é formada construindo-se o conjunto contendo elementos que pertencem a *A*, *B* ou ambos.

$$x \in (A \cup B) \leftrightarrow (x \in A \vee x \in B)$$

!

✓ Em sistemas gerenciadores de bancos de dados (SGBDs) é comum a existência de operações de união entre conjuntos de resultados a consultas ao banco de dados (***union***).

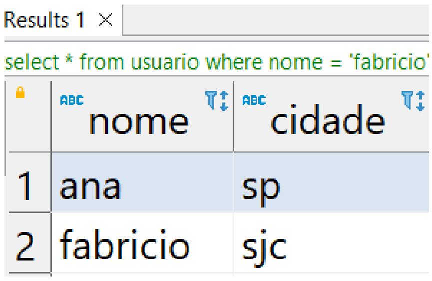
Exemplo:


usuario 1

select * from usuario

	nome	cidade
1	fabricio	sjc
2	ronaldo	bel
3	ana	sp
4	pedro	palmas

```
select * from usuario where nome = 'fabricio'  union  select  *
from usuario where cidade = 'sp';
```


Results 1

select * from usuario where nome = 'fabricio'

	nome	cidade
1	ana	sp
2	fabricio	sjc

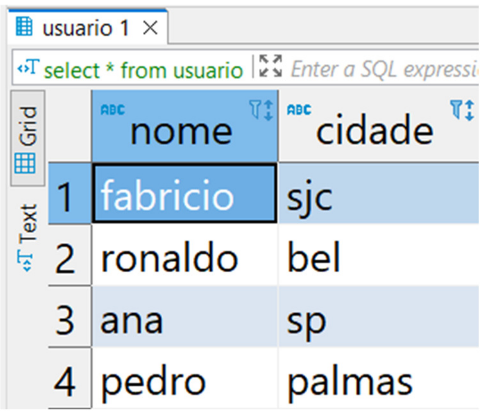
3.6.5. INTERSECÇÃO ENTRE CONJUNTOS

Considerando dois conjuntos A e B, a intersecção é formada construindo-se o conjunto dos elementos que pertencem a A **e** pertencem a B.

$$x \in (A \cap B) \leftrightarrow (x \in A \wedge x \in B)$$

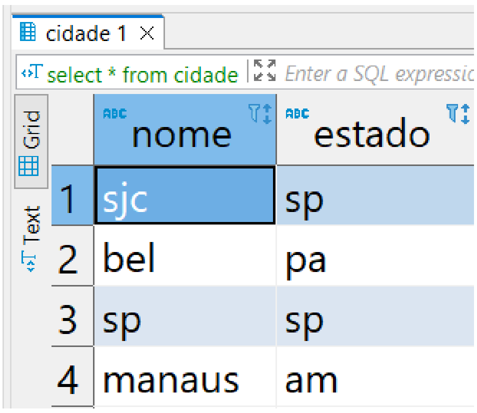
> **!**
>
> ✓ Em sistemas gerenciadores de bancos de dados (SGBDs) é comum a existência de operações de intersecção entre atributos comuns presentes em tabelas distintas (***joins***).

Exemplo:


usuario 1

select * from usuario

	nome	cidade
1	fabricio	sjc
2	ronaldo	bel
3	ana	sp
4	pedro	palmas


cidade 1

select * from cidade

	nome	estado
1	sjc	sp
2	bel	pa
3	sp	sp
4	manaus	am

```
select * from usuario as u join cidade as c on u.cidade =
c.nome;
```

	nome	cidade	nome	estado
1	fabricio	sjc	sjc	sp
2	ronaldo	bel	bel	pa
3	ana	sp	sp	sp

3.6.6. DIFERENÇA ENTRE CONJUNTOS

A diferença entre dois conjuntos *A* e *B* é definida como sendo os elementos que pertencem a *A* e não pertencem a *B*.

$$x \in (A - B) \leftrightarrow (x \in A \wedge x \notin B)$$

!

✓ Em sistemas gerenciadores de bancos de dados (SGBDs) é comum a existência de operações de diferença entre conjuntos que operam sobre o conjunto de resultados das consultas (*except*).

Exemplo:

	nome	cidade
1	fabricio	sjc
2	ronaldo	bel
3	ana	sp
4	pedro	palmas

```
select * from usuario except (select * from  usuario where
nome = 'fabricio');
```

Results 1 ×

select * from usuario except (select * from usu

	nome	cidade
1	ronaldo	bel
2	pedro	palmas
3	ana	sp

3.6.7. COMPLEMENTAR DE UM CONJUNTO EM RELAÇÃO A OUTRO

Considerando um subconjunto *A* de *B*, o complementar de *A* em relação a *B* é definido como sendo *(B – A)*.

3.6.8. PRODUTO ENTRE DOIS CONJUNTOS

O produto entre dois conjuntos *A* e *B*, conhecido como produto cartesiano e representado por $A \times B$, é formado por todos os pares (*a*, *b*), com $a \in A$ e $b \in B$.

$$A \times B = \{ (a,b) \mid a \in A \wedge b \in B\}.$$

Os pares *(a,b)* são chamados de pares ordenados ou duplas.

Para o caso do produto envolvendo N conjuntos tem-se as chamadas *n*-uplas ("ênuplas", por vezes também chamadas de "tuplas").

EXERCÍCIO RESOLVIDO

E.R.3.1. Exercício Resolvido: Para os conjuntos seguintes, determine: a)$A \cap B$; b) $A \cup B$; c) A-B; d) P(A)

$$A = \{ 1, 2, 3, 10\}$$

$$B = \{20, 1, 2, 3, -5\}$$

Resolução:

a)

A = { 1, 2, 3, 10}

B = {20, 1, 2, 3, -5}

$A \cap B$ ={ 1, 2, 3}

Marcar os elementos que se repetem em ambos os conjuntos auxilia na determinação do resultado das operações.

A intersecção é formada pelos elementos que aparecem em ambos os conjuntos.

b)

$A \cup B$ = {1, 2, 3, 10, 20, -5}

A união é formada pelos elementos que aparecem em um ou ambos os conjuntos.

c)

A-B = {10}

A diferença A-B é formada pelos elementos que pertencem a A mas não pertencem a B.

d)

P(A) = {∅, {1}, {2}, {3}, {10}, {1,2}, {1,3}, {1,10}, {2,3}, {2,10},{3,10}, {1,2,3}, {1,2,10}, {1,3,10} {2,3,10}, {1,2,3,10}}

O conjunto das partes de A deve incluir o conjunto vazio e todos os subconjuntos contendo 1, 2, 3 e 4 elementos (próprio A), formados a partir dos elementos de A.

EXERCÍCIOS

EP.4.1. Para os conjuntos A, B e C seguintes, determine o resultado das operações indicadas.

A = { 1, 2, 3, "moto"}

B= { "carro", "moto", 2}

a) $A \cup B$

b) $A \cap B$

c) A – B

d) B – A

e) P(B)

EP.4.2. Para os problemas seguintes, indique quais tipos numéricos são os mais adequados na hora de se implementar uma solução computacional (indique a qual dos conjuntos numéricos notáveis eles devem pertencer). Justifique sua resposta e forneça um exemplo para cada item.

a) Contagem do número de linhas de um arquivo de log.

b) Definição do preço, incluindo centavos, de mercadorias de uma loja.

c) Processamento de sinais de voz ou de comunicação em um sistema de engenharia de telecomunicações.

d) Marcação de temperaturas, em graus celsius, sem considerar frações de graus, em lugares de elevada variação de temperatura entre as estações do ano.

EP.4.3. Para ser elegível à participação em uma promoção, a idade do consumidor, subtraída de 18 deve ser superior a 5. Utilizando ***set builder notation***, descreva o conjunto das idades de todos os consumidores que podem participar dessa promoção.

EP.4.4. Se uma tabela em um banco de dados relacional, contendo ***n*** linhas cujas combinações de valores das colunas sejam únicas, pode ser considerada como um conjunto, diga qual a cardinalidade desse conjunto. Qual a cardinalidade do conjunto formado pelos conjuntos de resultado de todas as consultas possíveis? Ilustre sua resposta com um exemplo numérico simples através de uma tabela contendo 4 tuplas.

PC.4.1. Desenvolva uma classe, denominada de "Conjunto" que faça uso de um array para armazenar os elementos de um conjunto (i.e.: deve possuir o mesmo comportamento da interface Set do TypeScript. Adicionalmente, deve implementar os métodos denominados de inserir, pertence, união e intersecção. Implemente sua classe utilizando o arquivo denominado set_model.ts, fornecido pelo professor. Faça uso de tipos genéricos.

PC.4.2. Considere uma base de dados relacional que possua uma tabela de usuários com os seguintes valores:

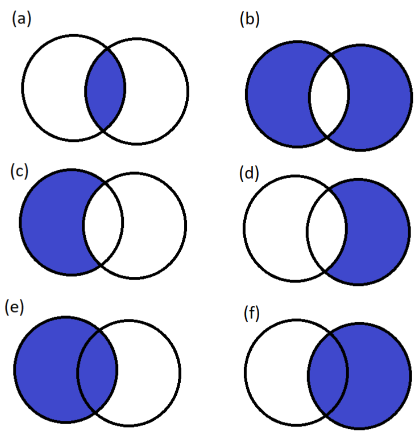
Nome	**Telefone**
"fabrício"	111
"beatriz"	222
"fabíola"	333

Utilizando consultas SQL, mostre que não é possível efetuar uma quantidade de consultas que seja superior a P(A), sendo A o conjunto formado pelas n-uplas que compõem as linhas da tabela. Para esse caso, ilustre todas as consultas e resultados possíveis.

PC.4.3. Considerando as seguintes tabelas contendo um cadastro de usuários e um cadastro de cidades, utilize consultas SQL, junções e operadores de conjuntos para ilustrar o resultado das operações mostradas nos diagramas de Venn.

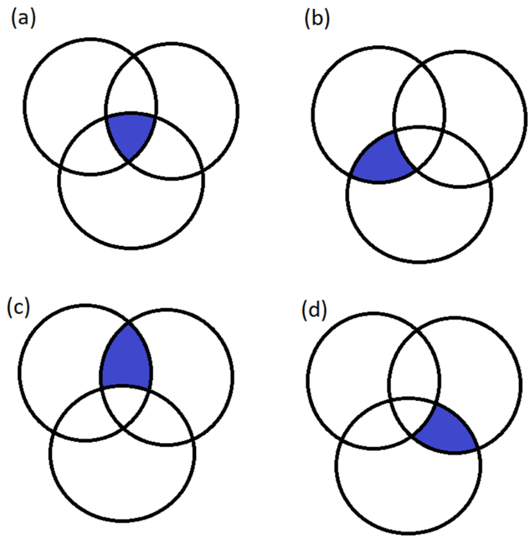
```
MariaDB [sets]> select * from usuario;
+----------+--------+
| nome     | cidade |
+----------+--------+
| fabricio | sjc    |
| thales   | sjc    |
| beatriz  | bel    |
| inacio   | palmas |
+----------+--------+
4 rows in set (0.000 sec)
```

```
MariaDB [sets]> select * from cidade;
+--------+-----------+
| cidade | regiao    |
+--------+-----------+
| bsb    | centro o. |
| bel    | norte     |
| sjc    | sudeste   |
+--------+-----------+
3 rows in set (0.000 sec)
```



PC.4.4. Repita o exercício anterior mas considerando a tabela adicional seguinte. Assegure-se de acrescentar valores às tabelas de forma que todas as regiões fiquem claramente ilustradas ao se efetuarem as consultas propostas:

```
MariaDB [sets]> select * from cidade_estado;
+--------+--------+
| cidade | estado |
+--------+--------+
| sjc    | sp     |
| bel    | pa     |
| bsb    | df     |
| manaus | am     |
+--------+--------+
4 rows in set (0.000 sec)
```



PC.4.5. Ilustre ao menos três diferentes tipos de Julia sets a partir do código-fonte fornecido pelo professor. Altere o esquema de coloração de modo a tornar a forma o mais próxima possível dos resultados mostrados em aplicações da Internet. Na sua resposta, inclua os valores iniciais para z e os demais parâmetros de ajuste da implementação do algoritmo.

PC.4.6. Utilizando a biblioteca ChartJS, adapte o gráfico de linhas ou dispersão XY para que exiba uma linha sólida desde a origem do sistema de coordenadas até as cordenadas do número complexo. Seu programa deve ser capaz de mostrar números complexos contendo magnitude menor ou igual a 10.