Colégio Técnico de Campinas
Departamento de Processamento de Dados

Cursos Técnicos Informática e Desenvolvimento de Sistemas

Matemática Aplicada para Algoritmos

Outro Mundo, de Mauritius Cornelius Escher (1898-1972)

1º Semestre - 2021

Algoritmo é um conjunto de ações que descrevem a solução de um problema.

Essas ações, executadas uma após a outra, de **forma ordenada e lógica,** geram a solução do problema apresentado.

A programação de computadores depende do uso de algoritmos como forma de organizar o raciocínio.

Esse raciocínio envolve uma dose alta de pensamento lógico. A Matemática é uma das melhores ferramentas que se pode utilizar para desenvolver o raciocínio lógico.

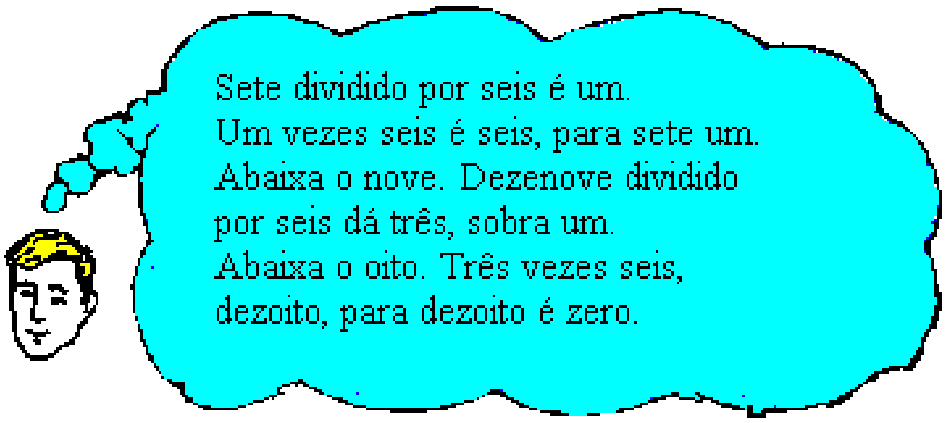
Há também vários algoritmos matemáticos que aprendemos no Ensino Fundamental e que servirão de base para o aprendizado de programação de computadores. Por isso, é importante revisarmos algumas técnicas matemáticas e (re)aprendermos algumas coisas interessantes e que facilitarão o desenvolvimento da lógica de programação.

O algoritmo tradicional da divisão

(http://educar.sc.usp.br/matematica/m4p2t6.htm)

Você já conhece este algoritmo:

```
798 |_6__
19    133
 18
  0
```

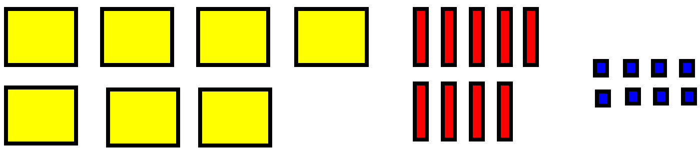


Trata-se de uma técnica para dividir que é, sem dúvida, bastante eficiente. Vamos discutir a compreensão da mesma.

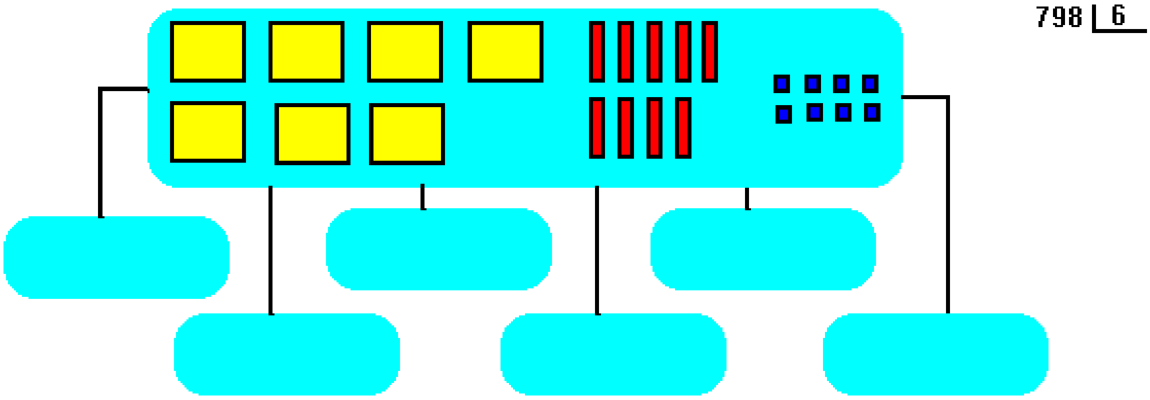
- Por que dividimos 7 por 6?
- Por que abaixamos o 9 e não o 98?
- Por que dizemos: 3 vezes 6 é 18, para 19 falta 1?

Para facilitar a compreensão do algoritmo usaremos materiais didáticos.

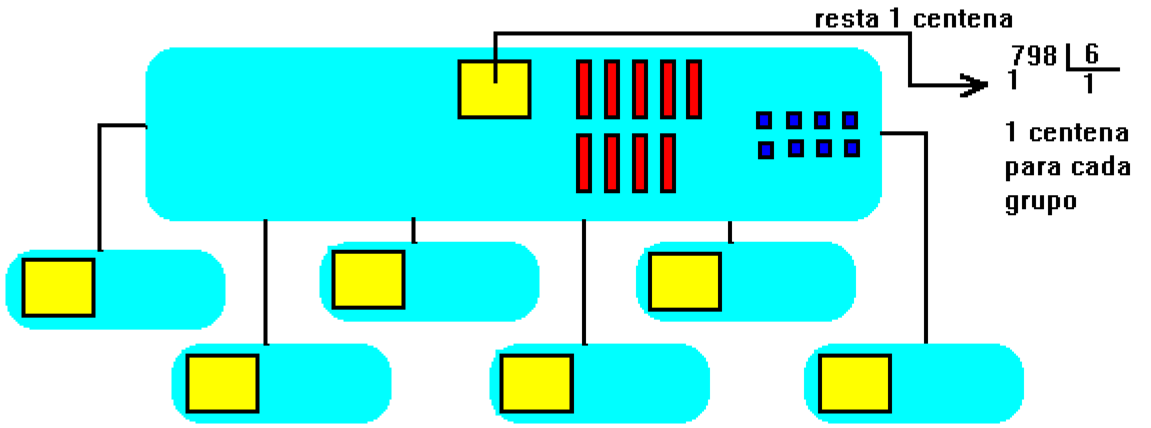
Vamos representar o número 798 com o material abaixo:



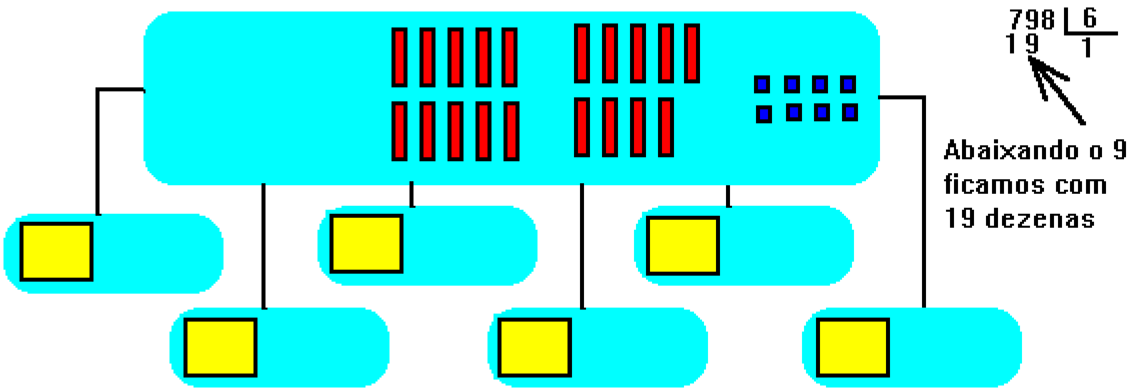
Para dividir 798 por 6 vamos distribuir igualmente 798 em 6 grupos:



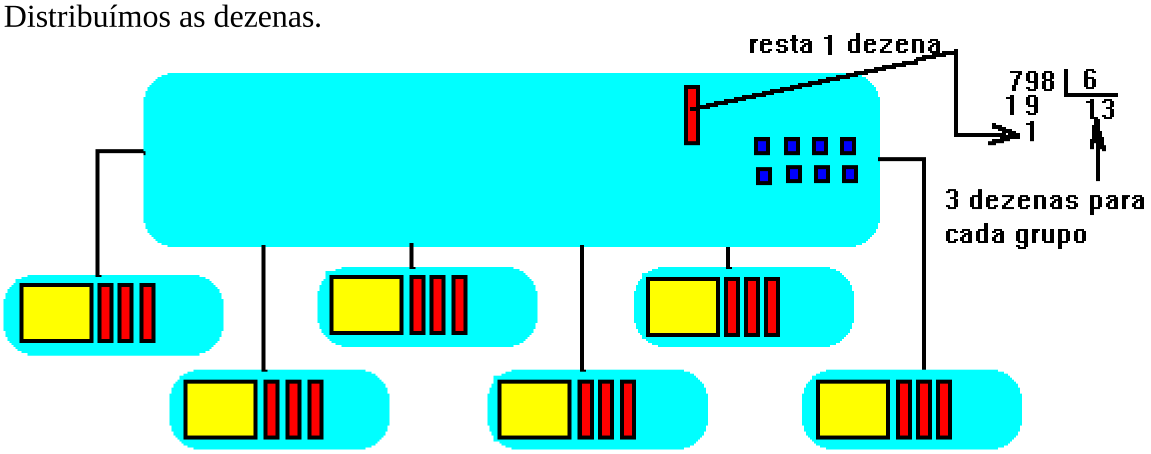
Começaremos distribuindo as centenas.



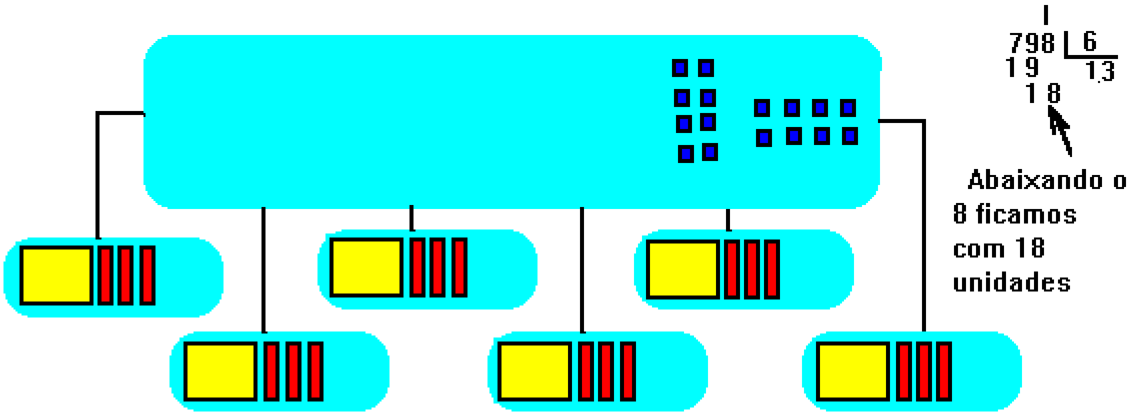
Desagrupamos a centena restante transformando-a em 10 dezenas. Agora temos 19 dezenas:



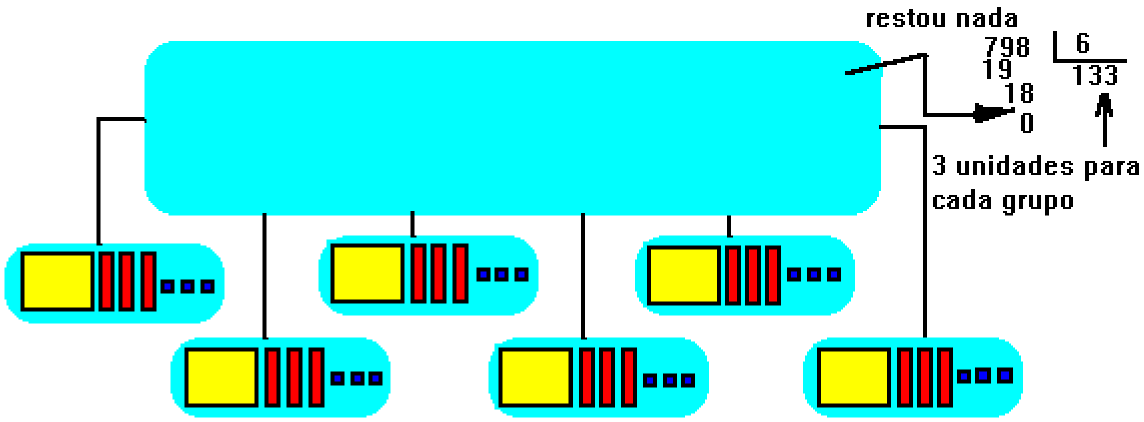
Distribuímos as dezenas.



Desagrupamos a dezena restante transformando-a em 10 unidades. Agora temos 18 unidades.



Finalmente distribuímos as unidades.



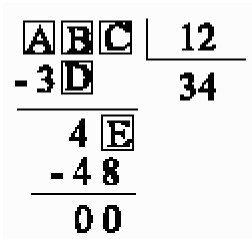
Em cada um dos 6 grupos temos 133 unidades. Esta divisão é exata, isto é, seu resto é zero.

A compreensão desse algoritmo da divisão depende da compreensão do nosso sistema de numeração, do domínio da subtração e de uma certa experiência com estimativas e cálculo mental.

No trabalho de sala de aula constatamos que a compreensão e o domínio desta técnica por parte dos alunos não é um processo simples.

MINIQUESTÃO 6

Complete os quadradinhos do algoritmo da divisão, representado abaixo, com os algarismos corretos:



A = ☐, B = ☐, C = ☐, D = ☐, E = ☐

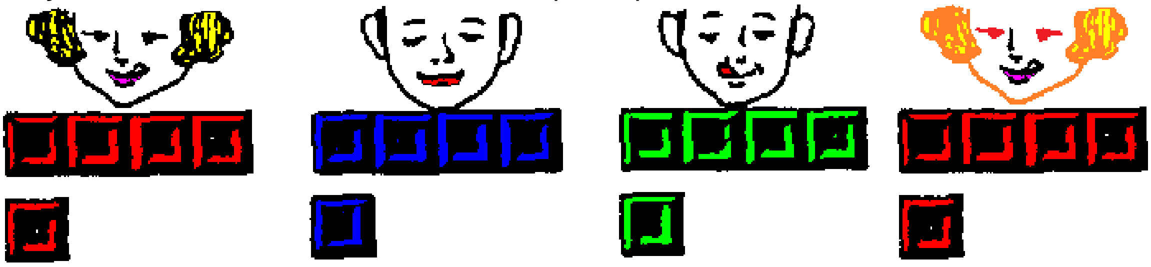
ENTRAR

LIMPAR

Nem tudo pode ser fracionado

http://educar.sc.usp.br/matematica/m4p1t3.htm

"Na semana passada ganhei do meu namorado cinco barras de chocolate. Chegando em casa resolvi dividir o chocolate entre meus quatro sobrinhos. Dei inicialmente uma barra para cada um e a barra restante dividi em quatro partes iguais. Deste modo, cada criança recebeu uma barra inteira e mais a quarta parte de uma barra de chocolate".



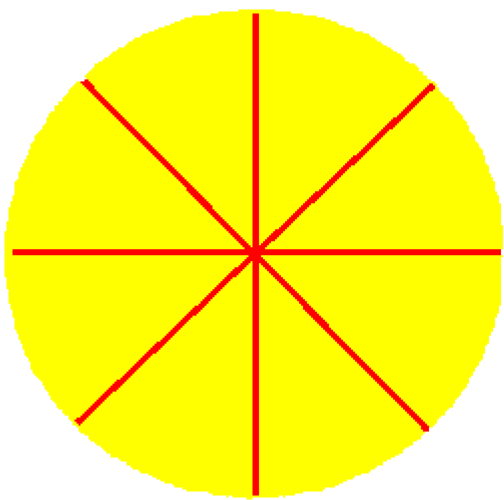
1,25 chocolate por guri

O fracionamento permitiu dividir igualmente cinco barras de chocolate entre as quatro crianças, de modo que não sobrasse chocolate. Veja agora esta outra situação:

"A gata Kiki, lá da vizinha, deu cria no portão da minha casa. A ninhada tem cinco filhotes. Prometi distribuir os gatinhos entre quatro crianças que moram nas redondezas. Como não quero privilegiar uma delas, presenteando-a com dois gatinhos, preciso decidir o que fazer com o quinto filhote".

Nesta situação, como o fracionamento não é possível, a divisão em partes iguais faz com que, necessariamente, sobre um gatinho (resto da divisão).

Às vezes é possível o fracionamento daquilo que se divide; às vezes não. É impossível fracionar gatos ou pessoas. Não faz sentido fracionar uma bola de futebol, uma boneca ou um automóvel. Mas pode-se fracionar o chocolate, uma pizza, uma porção de terra ou um círculo.



Círculo dividido em
8 partes iguais

As situações-problemas relacionadas com a divisão, nas quais é possível o fracionamento daquilo que está sendo dividido, conduzem-nos às frações, ao estudo dos números racionais.

As situações relacionadas com a divisão, nas quais não é possível o fracionamento daquilo que está sendo dividido, conduzem ao estudo da divisão no universo dos números naturais: 0, 1, 2, 3, 4, 5,...
As divisões efetuadas no universo dos números naturais são de dois tipos: as divisões que deixam resto (resto não nulo) e as divisões exatas (ou que têm resto zero). Por exemplo: a divisão de 10 por 4 deixa resto 2 e a divisão de 10 por 5 é exata.

Nos próximos dois itens vamos nos referir à divisão exata.

MINIQUESTÃO 3

Apresentamos abaixo algumas situações que envolvem divisões. Examine cada uma delas e verifique se o fracionamento é possível ou não e diga qual é o resto da divisão: (Escreva **sim** ou **nao** com **letras minúsculas**)

(a) Dividir um litro de leite entre 5 crianças.

(sim ou nao) resto =

(b) Dividir 6 laranjas entre 4 pessoas.

(sim ou nao) resto =

(c) Dividir 10 canetas entre 7 crianças.

(sim ou nao) resto =

(d) Dividir 5 bonecas entre 3 meninas.

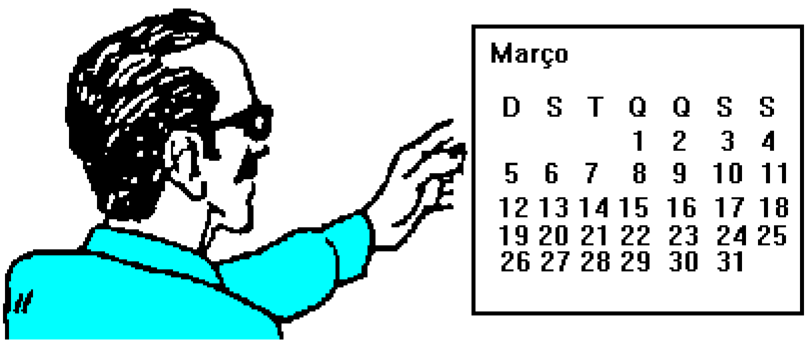
(sim ou nao) resto =

Dividendo, divisor, quociente e resto

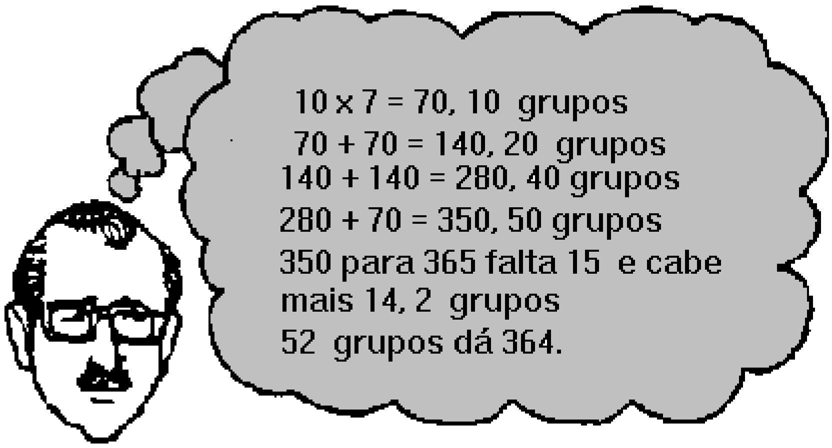
http://educar.sc.usp.br/matematica/m4p1t6.htm

Duas **situações-problema** nos ajudarão a construir alguns conceitos.

- **"Quantas semanas há em um ano?"**



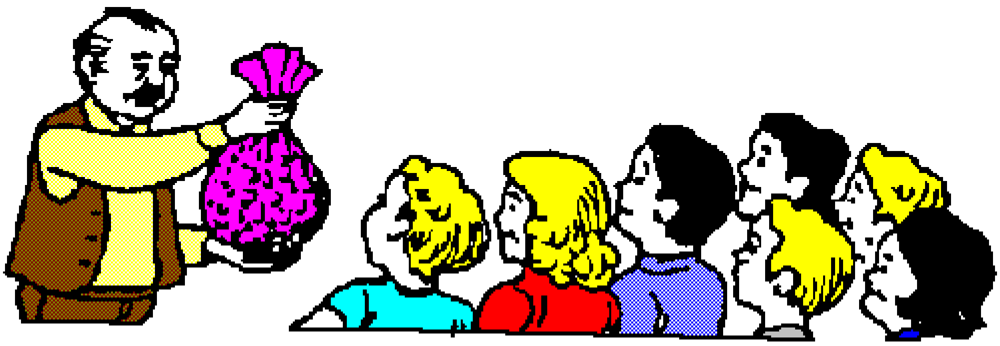
Um ano não bissexto tem 365 dias e a semana tem 7 dias. Queremos saber quantas semanas há em um ano, ou seja, quantos grupos de 7 há em 365. Este cálculo pode ser feito mentalmente.



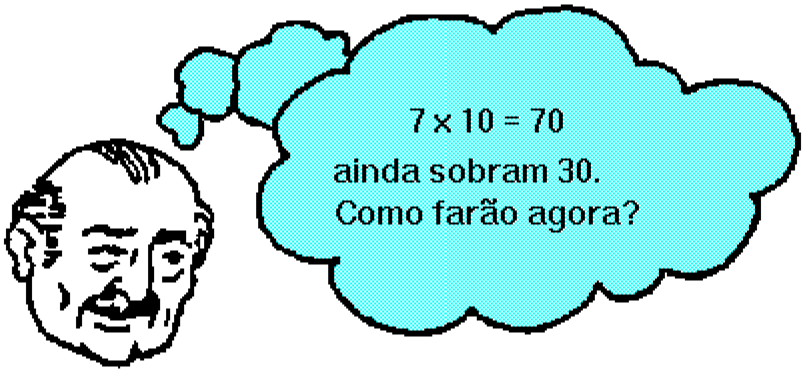
Como **365 = 7 x 52 + 1** , concluímos que um ano não bissexto tem 52 semanas e 1 dia. O problema proposto nos levou a uma divisão não exata. Esta divisão, que deixa resto 1, pode ser representada assim:

$$\begin{array}{r|l} 365 & 7 \\ \cline{2-2} 1 & 52 \end{array}$$

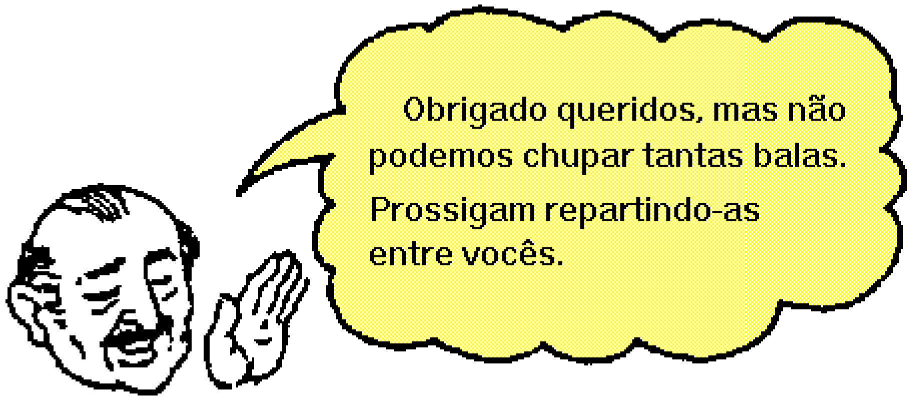
- **"Vovô Hermínio, que tem 7 netos, comprou 1 cento de balas. Sem dizer quantas balas havia no saco, entregou-o às crianças com a recomendação de que distribuíssem as balas igualmente entre elas."**



Sentadas no chão, formando uma roda, as crianças decidiram pegar 10 balas cada uma. O saco ia passando de mão em mão e cada uma, na sua vez, retirava suas balas. Vovô observava os netos.



Na segunda rodada as crianças decidiram pegar mais 3 balas cada uma. Isto feito, olharam as balas que ainda restaram no saco e as entregaram ao vovô, com a recomendação que as repartisse com a vovó.



Na terceira rodada cada neto pegou uma bala. As duas restantes ficaram para os avós.

Após a primeira rodada cada criança tinha 10 balas e restavam 30 no saco: **100 = 7 x 10 + 30.** Era possível prosseguir a distribuição. Após a segunda rodada cada uma tinha 13 balas e restavam 9 no saco: **100 = 7 x 13 + 9.** Nesse momento, apesar de ser possível ainda prosseguir, os netos deram por encerrada a distribuição. Mas o avô pediu que prosseguissem e, após a terceira rodada, cada um tinha 14 balas. Restavam 2 no saco: **100 = 7 x 14 + 2.**

Neste ponto, como **2 é menor do que 7**, e não havia a intenção de **fracionar** as balas, a divisão se encerrou.

As idéias presentes nas situações anteriores estão embutidas na definição de **divisão de números naturais.**

Dividir um número natural **a** pelo número natural **b** significa encontrar outros dois números naturais **q** e **r** que obedeçam a estas condições: **a = b x q + r** , e , **r < b (r é menor do que b)**.

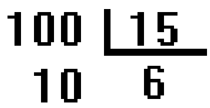
Representamos a divisão assim:

$$\begin{array}{l|l} a & b \\ \cline{2-2} r & q \end{array}$$

O número **a** chama-se **dividendo**, **b** é o **divisor**, **q** é o **quociente** e **r** é o **resto**.

EXEMPLOS:

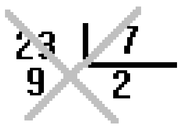
- Vejamos a divisão

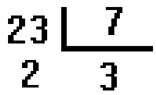


Como 100 = 15 x 6 + 10 , e , 10 < 15, dizemos que na divisão de 100 por 15 o quociente é 6 e o resto é 10.

- É verdade que **23 = 7 x 2 + 9**

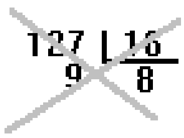
Entretanto **não é correto** afirmar que, na divisão de 23 por 7, o quociente é 2 e o resto é 9, pois 9 é maior do que o divisor 7 e, portanto, ainda podemos continuar a divisão.
A divisão correta é:

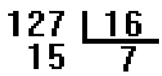




- A "divisão" abaixo está errada pois, apesar de 9 ser menor que 16, **não é verdade que** : **127 = 16 x 8 + 9**

A divisão correta é:





Nesta parte da lição abordamos uma série de conceitos e idéias relacionadas com a divisão. Na parte 2 veremos o cálculo mental, as propriedades e as técnicas de cálculo referentes a essa operação.

MINIQUESTÃO 6

Complete a tabela:

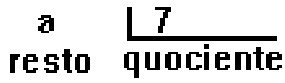
	dividendo	divisor	quociente	resto
a)	8	3		
b)		4	3	1
c)	22		3	
d)	17		3	2
e)	40		8	

ENTRAR

LIMPAR

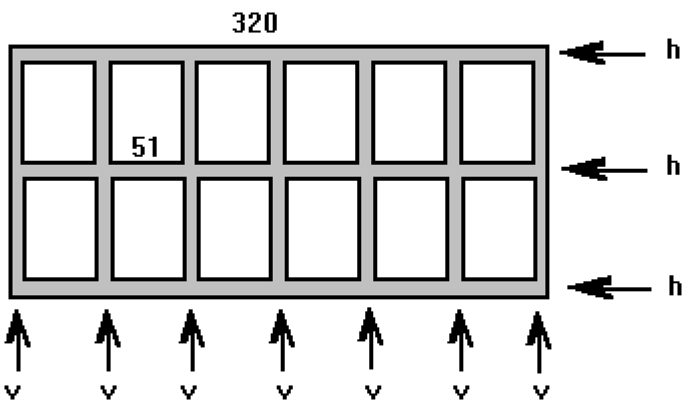
Exercícios

1. Considere a divisão do número natural **a** por **7**:



a)O resto dessa divisão pode ser 10?
b)Quais são os possíveis valores do resto dessa divisão?

2. Um painel mede 140 cm por 320 cm. Clélia deseja fixar 12 cartazes no painel. Os cartazes medem 64 cm por 51 cm. Como é muito caprichosa ela quer colocar os cartazes de modo que os espaçamentos verticais, (**v**), sejam iguais e os espaçamentos horizontais, (**h**), também sejam iguais.



Quais devem ser estes espaçamentos?

(a) **v** = ____ **cm**.

(b) **h** = **cm**.

3. As perguntas seguintes referem-se a esta divisão:

$$\begin{array}{r|l} 33 & 9 \\ \hline 6 & 3 \end{array}$$

a) Se o dividendo aumentar de 2 unidades de quanto aumentará o quociente?

O quociente aumentará de unidades.

b) Se o dividendo aumentar de 2 unidades de quanto aumentará o resto?

O resto aumentará de unidades.

c) Se o dividendo aumentar de 5 unidades de quanto aumentará o quociente?

O quociente aumentará de unidades.

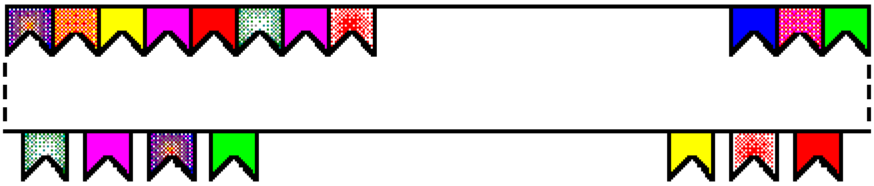
d) Se o dividendo aumentar de 5 unidades de quanto diminuirá o resto?

O resto diminuirá de unidades.

e) Se o divisor aumentar de 1 unidade de quanto aumentará o quociente?

O quociente aumentará de unidades.

4. Por ocasião das festas juninas Jussara e Raimundo prepararam as bandeirinhas para enfeitar a classe.



Raimundo colocou as bandeirinhas todas juntas, mas Jussara espaçou-as regularmente; ela mateve o espaço até mesmo nas extremidades. Os dois fios têm o mesmo comprimento (615 cm) e cada bandeirinha ocupa 15 cm de fio. Jussara colocou 7 bandeirinhas menos que Raimundo.

a) Quantas bandeirinhas Jussara colocou?

Jussara colocou

b) Qual é o espaçamento entre as bandeirinhas de Jussara?

O espaçamento é de cm cada um.

5. Num parque de diversões, a barraca de tiro ao alvo funciona no seguinte esquema: o freguês paga R$5,00 por 5 tiros e recebe R$3,00 por um tiro na "mosca" (centro do alvo). Miguel deu 20 tiros e saiu da barraca com R$16,00 a mais do que quando chegou.
Quantos tiros ele acertou na "mosca"?

Miguel acertou tiros na "mosca".

6. A seguir temos 4 sentenças incompletas. Você deve completá-las usando os sinais **+ - x : ( )** de modo a obter sentenças verdadeiras.

(a) 1 1 1 1 = 4

(b) 4 4 4 4 = 7

(c) 13 13 13 13 = 3

(d) 6 6 6 6 = 1296

Resposta:

7. Considere a divisão de um número natural n por 5:

$$\begin{array}{l|l} n & 5 \\ \cline{2-2} r & q \end{array}$$

As afirmações referem-se a esta divisão. Coloque no quadradinho a letra **V** (maiúscula) se a proposição é verdadeira e a letra **F** (maiúscula) se ela é falsa:

(a) ☐ **n = 5 x q + r** e **r < 5**

(b) ☐ Se **n** aumentar de 1 unidade então **q** aumentará, necessariamente, de 1 unidade.

(c) ☐ Se **n** aumentar de 1 unidade então **r** aumentará, necessariamente, de 1 unidade.

(d) ☐ Há somente **5** valores possíveis para **r**.

(e) ☐ Há somente **5** valores possíveis para **n**.

O zero, o um e as quatro operações

http://educar.sc.usp.br/matematica/m4p2t4.htm

Existem dois números que se comportam de maneira bastante especial com relação às quatro operações elementares. Estamos nos referindo ao **zero** e ao **um**.

São comuns estas opiniões sobre o zero:

Elas fazem sentido quando pensamos o zero associado à subtração. De fato, somando zero a um número ou subtraindo zero de um número obtemos sempre o próprio número. Estes fatos podem ser representados assim:

$$\mathbf{p + 0 = p}$$

$$\mathbf{p - 0 = p}$$

A letra p representa qualquer número

Entretanto, o papel do zero na multiplicação é bem diferente.

Veja:

5 x 0 = 0 + 0 + 0 + 0 + 0 = 0

0 x 3 = 3 x 0 = 0 + 0 + 0 = 0

De um modo geral:

$$a \times 0 = 0 \times a = 0$$

A letra a representa qualquer número

O zero como fator de uma multiplicação é "arrasador". Anula qualquer produto.

Vejamos agora o comportamento do zero na divisão. Lembremos que dividir **a** por **b** significa encontrar **c** de modo que c x b = a (estamos nos referindo à divisão exata).

Como dividendo o zero não oferece dificuldades. Por exemplo: 0 : 7 = 0 pois 0 x 7 = 0.

Agora vamos analizar um caso em que o zero é divisor. Por exemplo: dividir 2 por 0 é encontrar um número que multiplicado por 0 dê 2. Em outros termos:

se **2 : 0 = q** então **q x 0 = 2**

Sucede que não existe um número que multiplicado por 0 dê 2, pois todo número multiplicado por 0 dá 0. Logo, não existe o quociente da divisão de 2 por 0. Tal divisão é impossível.

Há ainda um caso a ser pensado: aquele em que o dividendo e o divisor são iguais a zero. Dividir 0 por 0 é encontrar um número que multiplicado por zero dê zero. Ora, todo número serve! Então haveria infinitos quocientes para a divisão de zero por zero. Esta situação criaria embaraços. Para a matemática, não há interesse algum em ter-se infinitos quocientes para uma só divisão. Por isso, também não se permite a divisão de zero por zero.

Moral da história: o zero nunca pode ser divisor!

Como já vimos, na adição o zero é neutro. Com relação à multiplicação, quem desempenha esse papel de neutralidade é o 1 uma vez que: **a x 1 = 1 x a = a**, qualquer que seja o número **a** (ou melhor, o número representado pela letra **a**).

Veja então que este carácter de neutralidade ou não do ***zero*** *e do* ***um*** *não é absoluto. Ele é relativo à operação considerada.*

A decomposição do dividendo: propriedade distributiva

Vamos interpretar o que a moça fez para dividir 1299 por 3. Ela decompôs 1299 na soma 1200 + 90 + 9. A seguir dividiu cada parcela da soma por 3;

1200 : 3 = 400; 90 : 3 = 30; 9 : 3 = 3.

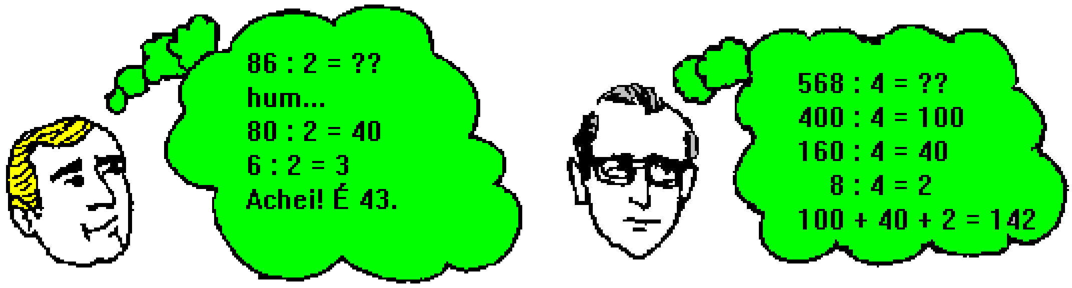
Depois somou os quocientes obtidos:

400 + 30 + 3 = 433.

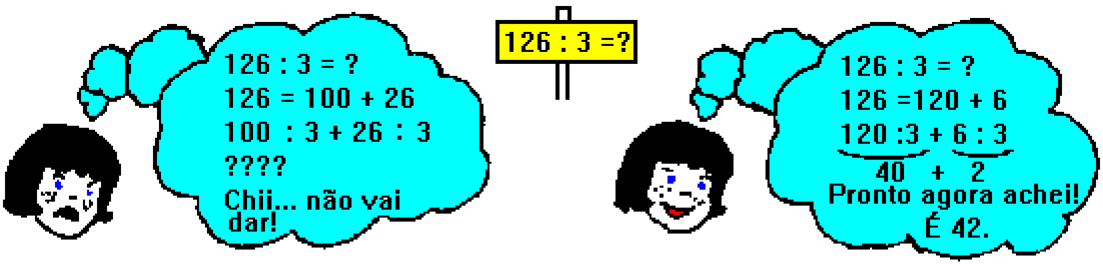
Podemos representar esta seqüência de cálculos assim:

1299 : 3 = (1200 + 90 + 9) : 3 = 1200 : 3 + 90 : 3 + 9 : 3 = 400 + 30 + 3 = 433.

As pessoas fazem decomposições como esta, com bastante naturalidade, quando calculam mentalmente.



A decomposição do dividendo não pode ser feita de qualquer jeito. Precisamos escolher bem os números.



As parcelas em que vamos decompor o dividendo devem ser divisíveis pelo divisor, ou seja, devem dar divisões exatas.

MINIQUESTÃO 2

Nome completo:

Seu e-mail:

Tente responder as duas questões abaixo sem usar o algoritmo da divisão, e sim, decompondo os números como no exemplo abaixo:

ex.: 424 : 4

424 = 400 + 20 + 4

400 : 4 = 100

20 : 4 = 5

4 : 4 = 1

portanto a divisão de 424 por 4 é exata, e dá 106 (100+5+1)

Responda V se verdade ou F se falso:

a) A divisão de 232 por 4 é exata?

b) A divisão de 500 por 45 é exata?

Nos exemplos anteriores vimos que:

- 1299 : 3 = (1200 + 90 + 9) : 3 = 1200 : 3 + 90 : 3 + 9 : 3
- 86 : 2 = (80 + 6) : 2 = 80 : 2 + 6 : 2
- 568 : 4 = (500 + 60 + 8) : 4 = 500 : 4 + 60 : 4 + 8 : 4
- 126 : 3 = (120 + 6) : 3 = 120 : 3 + 6 : 3

Vamos generalizar as idéias contidas nesses cálculos. Usaremos letras para números.

Consideramos a divisão exata do número **m** pelo número **n**. Vamos decompor **m** na soma das parcelas **a, b, c** de tal forma que **a, b** e **c** sejam divisíveis por **n** . Temos então:

$$m : n = (a + b + c) : n = a : n + b : n + c : n$$

Está é a propriedade distributiva da divisão exata. Ela vale para um número qualquer de parcelas do dividendo.

Vejamos outro exemplo:

$$\underbrace{(21 + 30 + 6 + 93 + 252)}_{402\,:\,3} : 3 = \underbrace{21 : 3}_{7} + \underbrace{30 : 3}_{10} + \underbrace{6 : 3}_{2} + \underbrace{93 : 3}_{31} + \underbrace{252 : 3}_{84}$$

$$402 : 3 = 7 + 10 + 2 + 31 + 84$$

$$134 = 134$$

MINIQUESTÃO 3

As letras **m**, **a** e **b** representam números quaisquer, sendo **a** e **b** diferentes de zero. Verifique se a igualdade é: **V** se verdadeira ou **F** se falsa.

☐ m : (a + b) = m : a + m : b

Dividir subtraindo

http://educar.sc.usp.br/matematica/m4p2t7.htm

No item anterior procuramos compreender o algoritmo tradicional da divisão. Há outras técnicas para dividir. Consideramos, por exemplo, a divisão de 17 objetos entre 5 pessoas.

Primeiro damos 1 objeto para cada pessoa. Como foram distribuídos 5 objetos, restam 17 - 5 = 12 objetos para serem distribuídos:

$$\begin{array}{r|l} 17 & 5 \\ \hline -5 & 1 \\ \hline 12 & \end{array}$$

1 objeto para cada pessoa

restam 12 objetos para distribuir

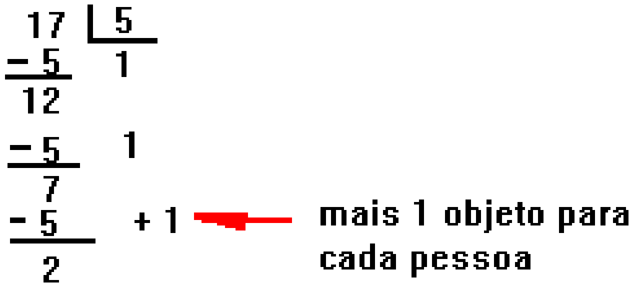
Damos mais um objeto para cada pessoa.

$$\begin{array}{r|l} 17 & 5 \\ \hline -5 & 1 \\ 12 & +1 \\ -5 & \\ \hline 7 & \end{array}$$

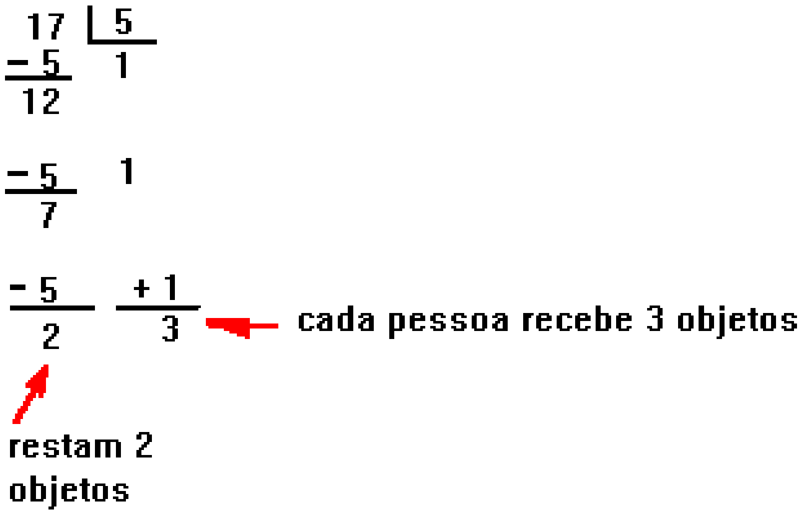
mais 1 objeto para cada pessoa

restam 7 objetos para distribuir

A seguir mais outro.

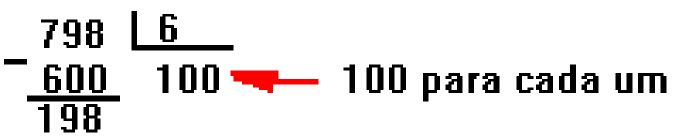


Como só restam 2 objetos, admitindo que não se deseja o fracionamento, a divisão está encerrada.

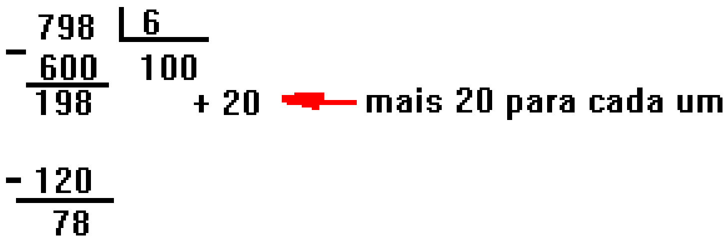


É claro que, comparado com algoritmo tradicional, este tem a desvantagem de ser mais demorado (contudo é possível "enxugá-lo"). Entretanto, este processo de subtrações sucessivas tem uma grande vantagem: ele é compreendido pelo aluno com muito mais facilidade.

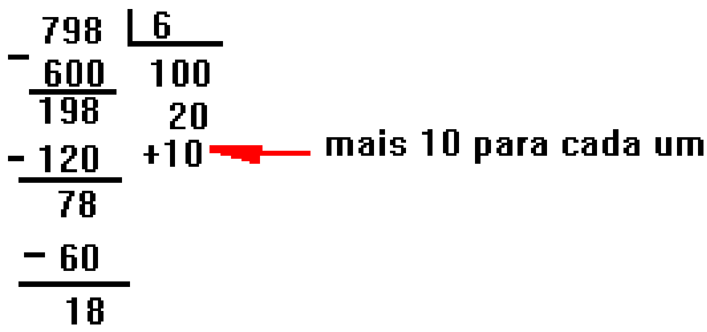
Vejamos outro exemplo: a divisão de 798 por 6. Neste caso, se fôssemos distribuindo de 1 em 1, o trabalho seria penoso! Fazendo uma estimativa, decidimos distribuir 100 para cada um dos 6:



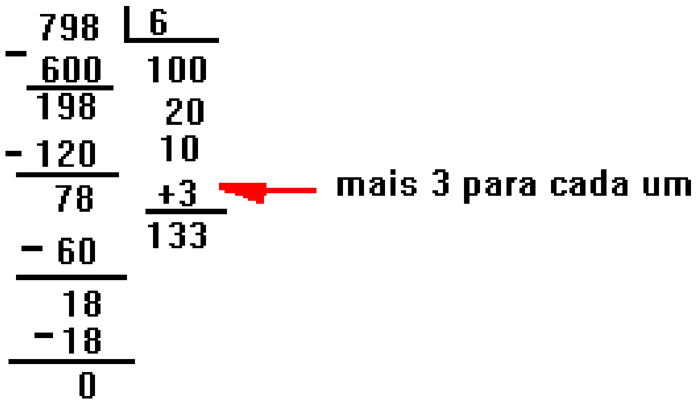
Nova estimativa e decidimos distribuir mais 20 para cada uma:



Agora mais 10 para cada um:



Finalmente:



Esta técnica é conhecida por algoritmo das subtrações sucessivas ou algoritmo americano.

Como já afirmamos, o aluno entende este algoritmo muito mais facilmente que o algoritmo tradicional.

Quanto ao fato de ser mais demorado, isto é relativo. Com a prática, fazendo estimativas e cálculos mentais, os alunos logo aprendem a "enxugá-lo". Estas considerações não têm a finalidade de sugerir que se use um e não o outro algoritmo. É interessante conhecer (e compreender!) várias técnicas para dividir.

Conhecendo vários processos e as dificuldades envolvidas na compreensão de cada um, deveremos decidir o que é mais adequado ao desenvolvimento do seu pensamento e à criação de algoritmos para computadores.

Explique como são feitas as divisões abaixo pelo algoritmo das subtrações sucessivas:

a) 540 : 12 =

b) 2244 : 17 =

Outras propriedades da divisão exata

Observe a tabela e procure descobrir que regularidade ela contém.

dividendo	divisor	quociente
6	2	3
12	4	3
18	6	3
24	8	3
30	10	3
36	12	3
42	14	3
48	16	3

A primeira linha correspondente à divisão exata 6 : 2 = 3. Da primeira para segunda linha, o dividendo e o divisor foram ambos multiplicados por 2; o quociente permaneceu inalterado.

Da primeira para a terceira linha, o dividendo e o divisor foram multiplicados por 3; o quociente é o mesmo. E assim por diante; cada divisão foi gerada a partir da primeira, multiplicando-se o dividendo e o divisor por um mesmo número. O quociente não muda. Generalizando, podemos representar esta propriedade assim:

se a : b = c, então (ma) : (mb) = c

Pelo que vimos as letras ***b*** e ***m*** não podem representar o número zero.

Agora observe esta outra tabela. Que regularidade ela contém?

dividendo	divisor	quociente
6	2	3
12	2	6
18	2	9
24	2	12
30	2	15
36	2	18
42	2	21
48	2	24

Note que o divisor é constante. Da primeira divisão para as seguintes, o dividendo foi multiplicado por 2, 3, 4, etc. Observe que o quociente correspondente também ficou multiplicado por 2, 3, 4, etc.

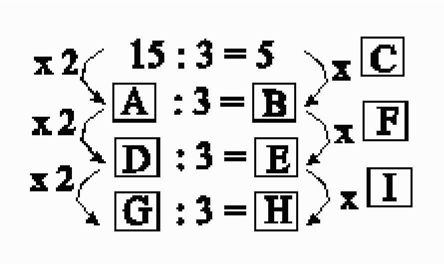
Vamos generalizar esta conclusão:

se a : b = c, então (ma) : b = (mc)

A letra b não pode representar o número zero.

MINIQUESTÃO 5

Preencha os espaços em branco de acordo com o desenho:



A = , B = , C = , D = , E = , F = , G = ,

H = , I =

Exercícios

1. Explique como se faz a divisão exata pelas duas maneiras diferentes propostas abaixo:

3360 : 32

a)pelo algoritmo tradicional:

b)pelo método das divisões sucessivas:

2. Este exercício também se refere às divisões exatas.Para resolvê-lo você poderá, se quiser, fazer as contas. Mas isso lhe dará muito trabalho. O exemplo seguinte poderá inspirá-lo, ajudando-o a encontrar as respostas rapidamente, sem a necessidade de grandes cálculos.
Exemplo: sabendo que 8008 : 88 = 91 é fácil concluir que 4004 : 44 = 91, pois 4004 é metade de 8008 e 44 é metade de 88.
Agora é sua vez.Complete os espaços em branco:

a) 2002 : 22 =

b) 1001 : 11 =

c) 2002 : 11 =

d) 4004 : 22 = $\square$

e) 8008 : 44 = $\square$

3. Neste exercício a expressão **qualquer que seja** m deve ser entendida como **qualquer que seja** o **número representado pela letra m.**
Escreva no quadradinho a letra **V** se a afirmação é verdadeira e **F** se ela é falsa.

a) $\square$ a : b = b : a, quaisquer que sejam a e b, com a e b diferentes de 0.

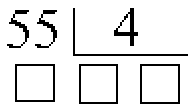
b) $\square$ (a + b) : c = a : c + b, quaisquer que sejam a, b e c diferente de 0.

c) $\square$ a : (b + c) = a : b + a : c, quaisquer que sejam a, b e c, com b e c diferentes de 0.

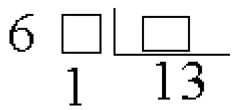
d) $\square$ (a + b) : c = a : c + b : c, quaisquer que sejam a, b e c, com c diferente de 0, a e b divisíveis por c.

e) $\square$ (a + b) : (c + d) = a : c + b : d, quaisquer que sejam a, b, c e d, com c e d diferentes de 0.

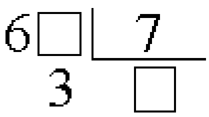
4. Conforme cada figura, preencha os quadradinho com o dígito correto:

a) 

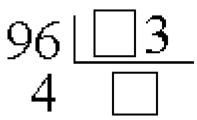
55 = 4 x $\square\square$ + $\square$

b) 

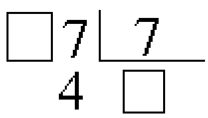
6$\square$ = $\square$ x 13 + 1

c) 

6$\square$ = 7 x $\square$ + 3

d) 

96 = $\square$3 x $\square$ + 4

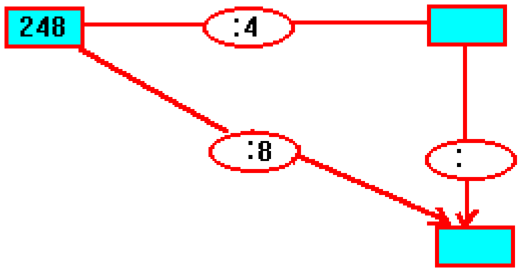
e) 

$\square$7 = 7 x $\square$ + 4

5. Observe a figura e preencha os espaços em branco:

Coloque primeiro o número maior;

"Dividir por 8 é o mesmo que dividir por ☐ e por ☐".



6. Observe a pontuação e dê o resultado:

a) 8 : (4 + 4) = ☐

b) 8 : 4 + 4 = ☐

c) 100 : (4 + 21) + 25 = ☐

d) 100 : (4 + 21 + 25) = ☐

e) 100 : 4 + 21 + 25 = ☐

7. Neste exercício, a letra **a** representa um número qualquer. Assim, afirmar que a + 0 = a significa afirmar que ***todo*** *número somado com 0 dá como resultado o próprio número*. Escreva no quadradinho a letra **V** se a afirmação é verdadeira e **F** se ela é falsa.

a) ☐ » a + 0 = a

b) ☐ » 0 - a = a

c) ☐ » a - 0 = a

d) ☐ » a x 0 = a

e) ☐ » a x 0 = 0

f) ☐ » a : 0 = a

g) ☐ » a : 0 = 0

h) ☐ » 0 : 1 = a

i) ☐ » 0 : a = 0, sendo a diferente de 0

Multiplicação

http://educar.sc.usp.br/matematica/mod3.htm
http://educar.sc.usp.br/matematica/m3p2t1.htm
http://educar.sc.usp.br/matematica/m3p2t4.htm
http://educar.sc.usp.br/matematica/m3p2t5.htm

Tabuada

2. Bases Numéricas

Quando duas pessoas tentam se **comunicar**, elas precisam usar uma forma de **comunicação** que as duas entendam, ou seja, uma **linguagem comum** às duas.

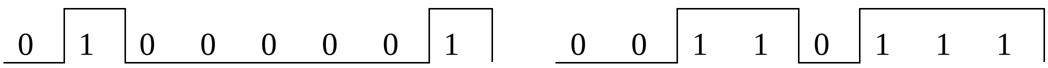
Na comunicação entre uma pessoa e um computador, o processo é semelhante.

O computador é uma máquina **construída** de tal forma que pode processar informações.

Tais informações são representadas dentro dele por meio de impulsos elétricos altos e baixos formando, assim, configurações de valores 1 (impulso alto) e de valores 0 (impulso baixo).

As configurações de 0 e 1 são **códigos** que representam as informações para o computador, da mesma maneira que as letras do alfabeto, quando agrupadas, podem representar informações na nossa linguagem. Essa linguagem de impulsos elétricos é a linguagem que um computador entende.

Estudaremos essa forma de representação no próximo capítulo, mas abaixo ilustramos desse conceito, indicando como a maioria dos computadores representa letras e dígitos. As linhas horizontais e verticais indicam a altura dos impulsos elétricos:



Letra ‘A’ (65 em decimal) Algarismo ‘7’ (55 em decimal)

Figura 1.1 – Representação em binário de informações alfanuméricas (letra ‘A’ e dígito ‘7’)

É óbvio que um ser humano não sabe usar esta linguagem de impulsos elétricos. Assim, devem-se procurar meios de traduzir a linguagem humana para a linguagem de máquina, usada por um computador. Isto é feito através do uso de **Linguagens de Programação.**

Com uma Linguagem de Programação, uma pessoa pode escrever **programas. Programas** são conjuntos de que ordens, dispostas em uma sequência lógica, para o computador **executar.**

Executar um programa significa que a sequência de ordens dispostas no programa será realizada pelo computador. Assim, é possível estabelecer contato com o computador, que é uma máquina capaz de receber instruções, efetuá-los e apresentar respostas ao seu operador.
Um programa é, portanto, um conjunto de instruções que são fornecidas a um computador para que ele as entenda e execute.

Um pouquinho de História

[copiado de http://pt.wikipedia.org/wiki/Sistema_binário_(matemática)]

O matemático indiano Pingala apresentou a primeira descrição conhecida de um sistema numérico binário no século III a.C..

Um conjunto de 8 trigramas e 64 hexagramas, análogos a números binários com precisão de 3 e 6 bits, foram utilizados pelos antigos chineses no texto clássico I Ching. Conjuntos similares de combinações binárias foram utilizados em sistemas africanos de adivinhação tais como o Ifá, bem como na Geomancia do medievo ocidental.

Uma sistematização binária dos hexagramas do I Ching, representando a sequência decimal de 0 a 63, e um método para gerar tais sequências, foi desenvolvida pelo filósofo e estudioso Shao Yong no século XI. Mas não há evidências que Shao Yong chegou à aritmética binária.

O sistema numérico binário moderno foi documentado de forma abrangente por Gottfried Leibniz no século XVIII em seu artigo "Explication de l'Arithmétique Binaire". O sistema de Leibniz utilizou 0 e 1, tal como o sistema numérico binário corrente nos dias de hoje.

Em 1854, o matemático britânico **George Boole** publicou um artigo fundamental detalhando um sistema lógico que se tornaria conhecido como Álgebra Booleana. Seu sistema lógico tornou-se essencial para o desenvolvimento do sistema binário, particularmente sua aplicação a circuitos eletrônicos.

Em 1937, Claude Shannon produziu sua tese no MIT que implementava Álgebra Booleana e aritmética binária utilizando circuitos elétricos pela primeira vez na história. Intitulado "A Symbolic Analysis of Relay and Switching Circuits", a tese de Shannon praticamente fundou o projeto de circuitos digitais.

Acredita-se que a necessidade de criação de números veio com a necessidade de contar. Seja o número de animais, alimentos, ou coisas do tipo. Como a evolução nos legou algumas características, como os cinco dedos em cada mão (fingers) e cinco dedos em cada pé (toes), seria muito natural que os primeiros sistemas de numeração fizessem uso das bases 10 (decimal) e 20 (vigesimal). O número 80, em francês, escrito como quatre-vingt (ou, quatro vezes o vinte) é remanescente de um sistema vigesimal.

(Copiado de http://www.inf.ufsc.br/ine5365/sistnum.html)

2.1. UTILIDADE DA BASE BINÁRIA

Citamos anteriormente que a linguagem interna de um computador é baseada em dígitos 0 e 1, formando o que se chama de linguagem de máquina ou linguagem binária.

Vamos agora estudar os fundamentos matemáticos dessa maneira de representar informações numéricas.

Esse estudo é importante para compreendermos como os dados são armazenados na memória de um computador, e o que ocorre internamente no momento de compilação de um programa fonte, escrito numa linguagem de programação, quando ele é convertido para a linguagem de máquina, que permite ao computador executar os comandos contidos no algoritmo descrito no programa.

O computador, como sabemos, é uma máquina e, portanto, diversas técnicas de engenharia e de teoria matemática foram aplicadas na sua concepção e construção, desde os primeiros modelos até os atuais.

Já que ele é composto por partes eletroeletrônicas, utiliza eletricidade para funcionar, e o próprio armazenamento de informações é baseado em eletricidade.

Esse fato é facilmente verificável pois, quando ocorre falta de energia, os dados armazenados na memória dos computadores são perdidos.

Nós, seres humanos, estamos familiarizados com a base **10** (decimal), no dia-a-dia, pois aprendemos, desde longínquas eras, a efetuar cálculos usando os dez dedos de nossas mãos. Dos dez dedos de nossas mãos surgiram os 10 dígitos (de 0 a 9), com que nos acostumamos a **representar** quantidades numéricas e a **realizar** operações aritméticas sobre essas quantidades.

Observe que a representação de uma quantidade numérica por meio de um conjunto de dígitos é uma abstração, pois poderíamos usar quaisquer símbolos para representar valores numéricos, e não apenas os dígitos. Portanto, um conjunto de dígitos apenas representa um valor numérico. Isso significa que o número não é a quantidade numérica concretamente, e sim, apenas e tão somente, uma representação, uma abstração dessa quantidade.

Já os computadores atuais trabalham exclusivamente com a base **2** (binário), assim é preciso fazer conversões entre estas bases quando se pretende inserir algum valor para ser processado pelo computador.

Mas por que isso ocorre? Como dissemos acima, o computador é um equipamento que utiliza eletricidade. Uma das maneiras de usar a eletricidade para representar algarismos (dígitos) é atribuir uma tensão elétrica (quantidade de volts) a cada dígito. Assim, poderíamos fazer o computador possui 10 tensões elétricas diferentes, uma para cada dígito, e usar sequências dessas tensões para representar os dígitos de um valor numérico. Por exemplo, vamos supor que cada dígito corresponda a um múltiplo de 5 volts, de forma que o dígito 0 (zero) corresponderia a 0 volts, o dígito 1 corresponderia a 5 volts, o dígito 2 corresponderia a 10 volts, e assim sucessivamente, enquanto o dígito 9 corresponderia a 45 volts.

Como exemplo, o valor 5128 seria representado pelas seguintes tensões elétricas, na memória de um computador desse tipo:

5 – □□□□□□ - 25 volts

1 – □□ - 5 volts

2 – □□□ - 10 volts

8 – □□□□□□□□□ - 40 volts

Para somar dois valores, por exemplo, o computador somaria as tensões de cada dígito correspondente nos dois valores armazenados na memória. As tensões que passassem de 45 volts seriam transferidas para serem somadas ao par de dígitos seguintes (algo como o “vai-um” da matemática):

Volt	Dígito															
45v	9															
40v	8															
35v	7															
30v	6															
25v	5															
20v	4															
15v	3															
10v	2															
5v	1															
0v	0															
		5	1	2	8	+	4	9	5	1	=	1	0	0	7	9

Isso pode parecer simples mas, na realidade, é bastante complicado implantar, num computador, a capacidade de trabalhar com 10 níveis de tensão elétrica diferentes. A sua complexidade seria muito maior, além de um maior consumo de energia.

A energia elétrica, em qualquer equipamento, varia. Quanto maior o nível de tensão elétrica num equipamento, maiores serão as flutuações e maiores as dificuldades de controlá-las. Portanto, ter 10 níveis diferentes de tensão elétrica é um convite para ter os dígitos mudando de tensão e, assim, provocar mudanças nos valores armazenados.

Assim, os engenheiros simplificaram esse projeto de equipamento, trabalhando com apenas duas tensões, baixa (0 volts, dígito 0) e alta (ao redor de 5 volts, dígito 1).

Isso foi possível não apenas porque os engenheiros assim queriam, mas com base numa teoria matemática, chamada Álgebra de Boole, que trabalha com apenas dois dígitos, 0 e 1, e com conceitos binários, como Falso e Verdadeiro e que, quando aplicada ao problema da criação de computadores digitais, embasou a solução de usar apenas dois níveis de tensão elétrica para representar os dígitos de qualquer valor numérico.

Assim, os valores armazenados na memória de um computador são representados usando o que chamados de Base 2, Base Binária, Sistema Numérico Binário ou, simplesmente, Binário. Ao invés de usar 10 níveis distintos de tensão elétrica para representar cada dígito, usamos apenas 2 níveis.

Você pode estar se perguntando: “Como isso é possível? Com dois dígitos eu posso apenas representar uma quantidade bem pequena de valores...”. Ou seja, para cada dígito, eu teria apenas duas possibilidades.

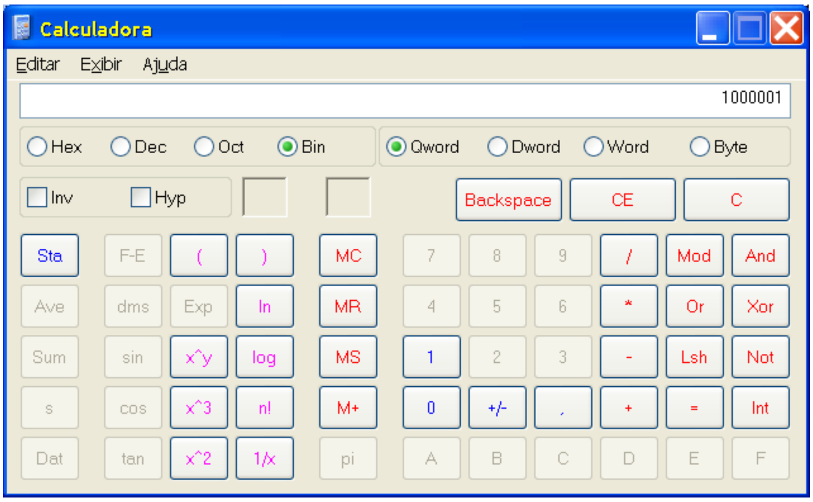
Esse pensamento decorre do fato de estarmos acostumados, desde crianças, a pensarmos em valores numéricos como sinônimo de algarismos decimais, Isso não é verdade, como dissémos acima, ao afirmar que o número não é a quantidade numérica, e sim, apenas, uma representação desta. Mas é normal pensar assim, já que, quando crianças, aprendemos a contar de 1 a 10, e aprendemos as teorias e as práticas matemáticas fundamentais usando a base 10, ou base decimal.

No computador, para representar cada dígito que usaríamos na base decimal, pode ser usado **mais de um** dígito binário.

Assim, o mesmo valor 5128 é representado, em binário, como 1010000001000.

Cada dígito binário no computador é chamado de **Bit**, que é uma sigla para Binary Digit, em inglês.

Imagine a calculadora do Windows. Nela, na visualização científica, existe um botão entitulado [Bin], que permite o cálculo usando valores representados em sistema binário.



Na figura acima, vemos que foi selecionado o sistema binário para os cálculos. Nesse momento, a calculadora mantém funcionais apenas os dígitos 0 e 1, sendo os de 2 a 9 desabilitados.

Mas se o computador sabe calcular apenas em binário, como é possível usar a calculadora com os dígitos no sistema decimal?

Afinal, a calculadora do Windows é também um programa e, assim, deveria usar a linguagem de máquina que o computador entende. Será necessário que usemos sempre os valores representados com os dígitos binários 0 e 1 em todos os programas?

Como pode ser facilmente verificado, isso não ocorre, já que quem já usou a calculadora deve tê-lo feito na base decimal, com os dígitos de 0 a 9 para representar os valores numéricos.

Obviamente que ninguém vai ficar convertendo números para o binário para então digitá-lo na calculadora e depois converter o resultado para decimal para usá-lo. Esse processo de conversão está, no caso da calculadora, pré-programado para ser feito por ela. O que deve ser entendido aqui é que, internamente, ela faz tudo em **binário**, em outras palavras: ela converte o que foi digitado para binário, faz o cálculo, converte o resultado para decimal e o apresenta.

A sequência de dígitos binários na figura acima corresponde à quantidade sessenta e cinto. Você pode confirmar essa afirmação selecionando o botão [Dec] da calculadora, que mudará o sistema numérico para o decimal e o visor apresentará os dígitos 6 e 5 que, como sabemos, representam o valor numérico sessenta e cinco. Sessenta e cinco, na base decimal, é representado pela sequência de dígitos decimais 65.

Portanto, você verifica que, ao usar a calculadora, ou a maioria dos programas em uso no mundo, os usuários digitam e leem valores na base decimal. Portanto, quando um valor decimal é digitado, o computador o converte para a base binária, realiza os cálculos e demais operações necessários com esse valor binário e, para exibir os resultados, estes são convertidos novamente para a base decimal, para só então aparecerem na tela e poderem ser lidos facilmente pelos usuários que, em geral, entendem bem o significado de sequências decimais que representam quantidades numéricas.

Mesmo na criação de programas, o programador usa, na linguagem de programação, valores decimais, que serão convertidos para binário durante a compilação do programa. Na prática de programação também podem ser utilizados a base **8** (octal), e a base **16** (hexadecimal), ambas derivadas da base 2 (note que estas bases facilitam a **digitação** somente, de qualquer forma ao ser compilado toda e qualquer base usada para escrever o programa é convertida para base 2 para que o valor seja usado pelo processador).

Valores numéricos representados em algumas bases			
10 (Decimal)	**2 (Binário)**	**8 (Octal)**	**16 (Hexadecimal)**

0	0	0	0
3	11	3	3
10	1010	12	A
15	1111	17	F
65	1000001	101	41
301	100101101	455	12D
1379	10101100011	2543	563
42685	1010011010111101	123275	A6BD

Repare como, na base maior (hexadecimal), o número de símbolos usados para representar o mesmo valor é bem menor que nas bases menores, é isso que facilita a digitação e memorização dos valores.

Repare também que no caso da simbologia da base haxadecimal são usadas algumas letras, isso ocorre porque temos símbolos para representar somente os algarismos de 0 a 9, como na base 16 é necessária a representação de algarismos de 10 a 15 então as letras de **A** até **F** são utilizadas para isso resultando na sequência: *0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F*.

Nos sistemas de numeração posicional, o valor do dígito em um número depende da posição que ele ocupa neste mesmo número.

1989 = 1000+900+80+9

1989 = $1x10^3 + 9x10^2 + 8x10^1 + 9x10^0$

Há um peso para cada posição ocupada pelo dígito. Os pesos crescem para esquerda na parte inteira e decrescem para a direita na parte fracionária

1989,4= $1x10^3 + 9x10^2 + 8x10^1 + 9x10^0+4x10^{-1}$

A representação posicional fornece uma forma simplificada para a escrita de números e permite a representação de qualquer número com um alfabeto (uma coleção de símbolos) restrito de dígitos.

Sistema posicional decimal	**b**ase R=10 Um alfabeto ordenado e 10 dígitos, {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}, e qualquer número pode ser representado com o uso deles.
Sistema posicional binário	base R = 2 alfabeto {0, 1}
Sistema posicional octal	base R = 8 alfabeto {0, 1, 2, 3, 4, 5, 6, 7}
Sistema posicional hexadecimal	base R = 16 alfabeto {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F}

Exemplos de Sistemas Não posicionais

- Sistema de Numeração Romano

No número XX, vinte em decimal, o valor do dígito X à esquerda é o mesmo daquele à direita. Neste caso a representação é aditiva, com X representando a quantidade decimal 10, e com a combinação XX associada a 10+10=20. Por outro lado em IX (nove em decimal) a representação é subtrativa.

- Outro exemplo: Sistema de Medição de tempo dividido em horas e minutos (uma espécie de base 60)

2.2. CONVERSÕES ENTRE BASES

A conversão entre bases pode ser realizada por meio de divisões sucessivas, uma técnica que funciona para qualquer combinação de bases, ou então, para os casos em que a base de origem e de destino pertencem à mesma base logarítmica, a conversão pode ser feita simplesmente por reagrupamento dos algarismos.

2.2.1. Divisões sucessivas

Neste método uma das bases tem que ser a decimal. Assim se nenhuma delas for decimal é necessário primeiro converter a base de origem para decimal e então converter para base de destino. Isso ocorre porque usaremos os 10 algarismos decimais (0 a 9) para representar valores.

Tomemos o exemplo da conversão do número 745, da base 10 (decimal) para a base 4. Uma série de ***divisões inteiras*** pela nova base (4) é realizada **repetitivamente até que** o resultado zere. Cada resultado de divisão (diferente de zero) será usado como dividendo na próxima divisão. A sequência dos **restos das divisões inteiras** é a sequência de algarismos da base de destino. Como a base de origem é decimal podemos usar o método diretamente:

- $745/4 = 186 \rightarrow 1$
- $186/4 = 46 \rightarrow 2$
- $46/4 = 11 \rightarrow 2$
- $11/4 = 2 \rightarrow 3$
- $2/4 = 0 \rightarrow 2$

Portanto $\mathbf{745_{10} = 23221_4}$

Conversão de 5128_{10} para a base 5

```
5128 |_5
   3  1025 |_5
          0  205 |_5
                0  41 |_5
                      1  8 |_5
                         3  1 |_5
                            1  0
```

Após calcular os quocientes e os restos das divisões, montamos o número convertido desde o último resto até o primeiro, resultando em 131003, que é o valor 5128 (na base 10) expresso na base 5.

Portanto, podemos dizer que $131003_5 = 5128_{10}$.

Mas como provamos que essas operações são verdadeiras? Em outras palavras, como fazemos a operação inversa, de modo que 131003 na base 5 seja convertido para 5128 na base 10? Bem, um número inteiro é representado na base 10 através da multiplicação de seus dígitos por potências de 10. Cada

$5 \times 10^3 + 1 \times 10^2 + 2 \times 10^1 + 8 \times 10^0$

```
    8+
   20+
  100+
 5000=
 5128
```

dígito da representação um número inteiro na base 10 possui um **peso**, que corresponde a uma potência de 10. Por exemplo, o próprio valor 5128 é representado na base 10 por

$8 \times 10^0 + 2 \times 10^1 + 1 \times 10^2 + 5 \times 10^3$

$8 \times 1 + 2 \times 10 + 1 \times 100 + 5 \times 1000$

$8 + 20 + 100 + 5000 = 5128$

Note que cada dígito é multiplicado por uma potência de 10. O expoente dessa potência é definido de acordo com a posição em que o dígito está. Assim, o primeiro dígito (a direita para esquerda) é multiplicado pela potência 10 elevado a 0, o segundo dígito é multiplicado pela potência 10 elevado a 1, o terceiro dígito é multiplicado por 10 elevado a 2, e assim sucessivamente. Podemos definir uma fórmula geral, para um valor de n dígitos, que seria semelhante a:

$$\text{Valor} = d_0 \times 10^0 + d_1 \times 10^1 + d_2 \times 10^2 + \ldots + d_{n-1} \times 10^{n-1}$$

Cada dígito recebeu um peso, dado pela sua posição na composição da representação do valor desejado. Isso é chamado de notação posicional.

Da mesma maneira, para verificarmos qual a representação de 131003_5 na base 10, usamos potências de 5 multiplicadas por cada dígito que forma o número.

$3 \times 5^0 + 0 \times 5^1 + 0 \times 5^2 + 1 \times 5^3 + 3 \times 5^4 + 1 \times 5^5 =$

$3 \times 1 + 0 \times 5 + 0 \times 25 + 1 \times 125 + 3 \times 625 + 1 \times 3125 = 3 + 0 + 0 + 125 + 1875 + 3125 = 5128_{10}$

Até aqui, usamos os 10 dígitos da base decimal para calcular as representações em outras bases. Podemos notar que a base 5 usará os dígitos de 0 a 4. Isso ocorre porque, para fazer a mudança de base, os dígitos na nova base são os restos das divisões feitas pelo valor da base. No caso da base 5, ocorrem divisões por 5, e os restos de qualquer divisão por 5 só podem ser 0, 1, 2, 3 e 4. Nunca haverá restos maiores ou iguais a 5.

Isso ocorre para qualquer número inteiro. Por esse motivo é que restos de divisões por 10 somente geram valores de 0 a 9. E, portanto, são esses os dígitos usados na base 10. De forma semelhante, podemos dizer que, para representar valores numéricos, a base 3 usa os dígitos 0, 1 e 2; a base 4 usa os dígitos 0, 1, 2 e 3; e a base 2 (binário) usa os dígitos 0 e 1, pois são estes os restos possíveis na divisão por 2.

Portanto, para convertemos 5128 para a base 2, fazemos diversas divisões por 2, encontrando o resto de cada uma. O resultado inteiro de uma divisão será usado como o dividendo da divisão seguinte:

5128	/ 2 =	2564	com resto 0
2564	/ 2 =	1282	com resto 0
1282	/ 2 =	641	com resto 0
641	/ 2 =	320	com resto 1
320	/ 2 =	160	com resto 0
160	/ 2 =	80	com resto 0
80	/ 2 =	40	com resto 0
40	/ 2 =	20	com resto 0
20	/ 2 =	10	com resto 0
10	/ 2 =	5	com resto 0
5	/ 2 =	2	com resto 1
2	/ 2 =	1	com resto 0
1	/ 2 =	0	com resto 1

Quando o resultado de uma divisão é zero, terminamos o processo de conversão.

Em seguida, montamos o número convertido, na ordem inversa do cálculo dos restos. Assim, o dígito mais à esquerda será o último resto calculado, e o dígito mais á direita será o primeiro resto calculado.

Sendo assim, 5128_{10} é igual a 1010000001000_2.

Isso pode ser comprovado facilmente, pois

$5128 = 1 \times 2^3 + 1 \times 2^{10} + 1 \times 2^{12} = 8 + 1024 + 4096$

Expoente	12	11	10	9	8	7	6	5	4	3	2	1	0
Dígitos	1	0	1	0	0	0	0	0	0	1	0	0	0

As potências de 2, com expoentes 0, 1, 2, 4, 5, 6, 7, 8, 9 e 11, foram multiplicadas pelo dígito 0, de forma que não afetam a soma acima.

Usamos nesta discussão, até o momento, apenas bases menores que 10, que usam, para representar os valores, os dígitos decimais (de 0 a 9).

Mas e no caso de bases maiores que 10, com a base 16, muito usada em computação? Como representar os dígitos, se a base 16 gera restos de 0 a 15 nas divisões?

Nesses casos, temos que usar letras para calcular os resultados, supondo que os valores de resto maiores que 10 correspondam a letras do alfabeto. Assim, o resto de divisão 10 será representado como A, o resto 11 será representado por B, e assim sucessivamente. No caso da base 16, teremos a seguinte relação:

10 = A   11 = B   12 = C   13 = D   14 = E   15 = F

Na base 18, além dos valores acima, teríamos 16 = G e 17 = H. Em outras bases, teremos o mesmo procedimento.

Outro exemplo: conversão de $4C_{18}$ para a base 7:

Como o valor de origem está na base 18 primeiro precisamos convertê-lo para a base 10:

$4C_{18} = 4 \times 18^1 + 12 \times 18^0 = 72 + 12 = 84_{10}$

Repare na substituição do *algarismo* **C** pelo valor decimal 12. Isso foi feito porque como na base 18 precisamos que cada algarismo represente 18 valores diferentes, comumente se usa letras na sequência do último algarismo numérico para o qual temos um símbolo, o **9**, então o *A* vale 10, o *B* vale 11 e assim sucessivamente até o valor 17 (com o símbolo zero levado em consideração teremos então os 18 que precisávamos).

Agora sim aplicamos as divisões:

- $84/7 = 12 \rightarrow 0$
- $12/7 = 1 \rightarrow 5$
- $1/7 = 0 \rightarrow 1$

Assim: $\mathbf{4C_{18} = 84_{10} = 150_7}$

Mais um exemplo: converter 652_8 para a base 3:

$652_8 = 6 * 8^2 + 5 * 8^1 + 2 * 8^0 = 384 + 40 + 2 = 426_{10}$

- $426/3 = 142 \rightarrow 0$
- $142/3 = 47 \rightarrow 1$
- $47/3 = 15 \rightarrow 2$
- $15/3 = 5 \rightarrow 0$
- $5/3 = 1 \rightarrow 2$
- $1/3 = 0 \rightarrow 1$

Assim: **$652_8 = 426_{10} = 120210_3$**

2.2.2. Reagrupamento

Quando as bases envolvidas são da mesma base logarítmica (resultados de potências) então a conversão pode ser facilmente feita por simples reagrupamentos dos algarismos e uso de pequenas tabelas de conversão. Por exemplo, entre as bases 16 e 8 ou entre 2 e 16 ou ainda entre as bases 27 e 9.

Na prática é muito usada a conversão entre as bases 2, 8 e 16 pelos motivos citados anteriormente. Segue uma tabela básica para estas conversões:

Tabela de conversão de bases de origem binária

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Binário	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Octal	0	1	2	3	4	5	6	7								
Hexadecimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Convertendo 111010110_2 para a base 16:

Pela tabela vemos que para cada algarismo em hexadecimal são necessários 4 algarismos para realizar sua representação em binário. Então o primeiro passo é separar o valor em base 2 em blocos de 4 algarismos:

111010110_2 = 1.1101.0110

Depois, consultando a tabela convertemos o valor de cada bloco para seu equivalente hexadecimal, assim teremos:

$111010110_2 = 1.D.6_{16} = 1D6_{16}$

Convertendo 111010110_2 para base 8:

Pela tabela vemos que para cada algarismo em octal são necessários 3 algarismos para realizar sua representação em binário. Então devemos separar o valor em base 2 em blocos de 3 algarismos:

111010110_2 = 111.010.110

Depois, consultando convertemos o valor de cada bloco para seu equivalente octal, assim teremos:

$111010110_2 = 7.2.6_8 = 726_8$

Finalmente uma conversão do valor $3A8_{16}$ para octal:

Primeiro convertemos para os blocos binários equivalentes com 4 dígitos:

$3A8_{16} = 3.A.8_{16} = 0011.1010.1000_2 = 1110101000_2$

Agora reagrupamos em blocos de 3 dígitos:

$1110101000_2 = 1.110.101.000_2 = 1.6.5.0_8$

Assim: **$3A8_{16} = 1650_8$**

Binários a decimais

Dado um número N, binário, para expressá-lo em decimal, deve-se escrever cada número que o compõe (bit), multiplicado pela base do sistema (base = 2), elevado à posição que ocupa. Uma posição à esquerda da vírgula representa uma potência positiva e à direita, uma potência negativa. A soma de cada multiplicação de cada dígito binário pelo valor das potências resulta no número real representado. Exemplo:

111011(binário)

$1 \times \mathbf{2^5} + 1 \times \mathbf{2^4} + 1 \times \mathbf{2^3} + 0 \times \mathbf{2^2} + 1 \times \mathbf{2^1} + 1 \times \mathbf{2^0} = 59$

Portanto, 111011 é 59 em decimal

2.2.3. Operações com binários

Soma de Binários

0+0=0
0+1=1
1+0=1
1+1=10, ou seja 0 e vai 1* (para somar ao digito imediatamente à esquerda)

Temos um exemplo de soma abaixo:

```
    *
    1100
+    111
  -----
= 10011
```

Explicando: Os números binários são base 2, ou seja, há apenas dois algarismos: 0 (zero) ou 1 (um). Na soma de 0 com 1 o total é 1. Quando se soma 1 com 1, o resultado é 2, mas como 2 em binário é 10, o resultado é 0 (zero) e passa-se o outro 1 para a "frente", ou seja, para ser somado com o próximo elemento, conforme assinalado pelo asterisco,como no exemplo acima. outro exemplo:

```
   **
    1100
+   1111
  -----
= 11011
```

Explicando: No caso acima, na quarta coluna da direita para a esquerda, nos deparamos com uma soma de 1 com 1 mais a soma do 1 (*) que veio da soma anterior. Quando temos esse caso (1 + 1 + 1), o resultado é 1 e passa-se o outro 1 para frente.

Exercícios

Obtidos em http://www.estv.ipv.pt/paginaspessoais/maeb/pc/Praticas_PDF/f.01.sistemas de numeração.pdf

1. Converta os seguintes números na base indicada para a base 10, usando uma das regras anteriores:

a) $(1011001100101)_2 = (?)_{10}$ b) $(7532123)_8 = (?)_{10}$
c) $(9BF2A2)_{16} = (?)_{10}$ d) $(24104331)_5 = (?)_{10}$
e) $(JJJJ)_{20} = (?)_{10}$ f) $(1001,1101)_2 = (?)_{10}$
g) $(3042,12)_8 = (?)_{10}$ h) $(3AA,BA)_{16} = (?)_{10}$

2. Converta os seguintes números na base 10 para a base indicada, usando uma das regras anteriores:

a) $(275)_{10} = (?)_2$ b) $(2756875)_{10} = (?)_2$
c) $(753296875)_{10} = (?)_8$ d) $(12341875)_{10} = (?)_{16}$

3. Execute as seguintes conversões:

a) $(1011011011)_2 = (?)_8$ b) $(100011101011)_2 = (?)_{16}$
c) $(101100010111)_2 = (?)_8$ d) $(F905C)_{16} = (?)_2$
e) $(7360214)_8 = (?)_2$ f) $(604237)_8 = (?)_{16}$
g) $(1011011011)_2 = (?)_{16}$ h) $(A08F31)_{16} = (?)_8$
i) $(8C2)_{16} = (?_2$ j) $(BA5)_{16} = (?)_8$

4. Execute as seguintes operações aritméticas:
a) $(01011011)_2 + (01100011)_2 = (?)_2$
b) $(01010001)_2 + (01000111)_2 = (?)_2$
c) $(1010001)_2 + (1000111)_2 + (11110111)_2 = (?)_2$
d) $(1010001)_2 + (1000111)_2 + (11110111)_2 + (10100101)_2= (?)_2$

5. Responda às seguintes questões:

a) Quantos números diferentes podem ser representados com 8, 16 e 32 símbolos em binário?
b) Considerando valores positivos e negativos, quais os valores mínimo e máximo que podem ser representados com 8, 16 e 32 bits?
c) Supondo que necessita de guardar uma quantidade que pode tomar valores entre 0 e 1024 e que dispõe de dispositivos de armazenamento com tamanho fixo de 8, 16 ou 32 bits. Qual a opção que escolheria?

6. O CD é um formato que se encontra largamente difundido como suporte de música, para a instalação de software e como suporte de informação. As razões do seu sucesso devem-se à sua imunidade ao ruído devido ao facto de usar a informação em formato digital e à sua elevada capacidade de armazenamento.

A informação no CD é armazenada da seguinte forma: numa fracção de segundo é guardada informação usando palavras de 16 bits; em cada segundo de música é guardada a informação relativa a 44100 fracções de segundo; são guardados dois canais de som independentes. Então, quantos bytes terá um CD de uma hora?

2.3. REPRESENTAÇÃO DE INSTRUÇÕES

A mudança de base e os números binários são usados para representar valores internamente na memória de computadores.

No entanto, no aprendizado desta disciplina, essas operações são realizadas pelo próprio computador, convertendo valores decimais que foram digitados pelo usuário para seus correspondentes em binário para, depois disso, efetuar operações de processamento desses dados, usando a base binária para isso. Existem circuitos eletrônicos que foram projetados e construídos para somar dois valores, por exemplo. Os valores em binário são armazenados na memória, são levados para esse circuito (usando o que se chama de barramento de memória) e o circuito realiza a operação de soma. O resultado é novamente levado para a memória, em binário. Se for necessário exibir o resultado, o próprio computador converterá o valor em binário para a sua representação na base 10, de forma que leremos o valor calculado em decimal, o que facilitará bastante a nossa interação com o computador.

Em outras palavras, na maioria das vezes o conhecimento da base binária ou de conversões entre bases numéricas não será necessário para desenvolvermos programas de computadores.

Mas, por outro lado, para aprendermos profissionalmente o desenvolvimento de programas, é importante possuir noções do que acontece internamente ao computador, principalmente quando percebemos que o computador não entende a nossa forma de expressar informações, e que não entendemos (normalmente) a forma como ele armazena informações.

Para você ter uma ideia, até mesmo as cores em um monitor de vídeo são representadas como números binários. A composição de 3 valores numéricos entre 0 e 255 informa a quantidade de vermelho, de verde e de azul que uma cor terá. Combinando as diversas possíbilidades dessas 3 cores, consegue-se gerar quase 17 milhões de cores diferentes.

A conversão de binário para a nossa maneira de representar informações (numéricas e alfabéticas) é feita pelo computador, mas também isso foi programado nele, não “nasceu” com ele.

Nos primeiros computadores digitais, os programadores tinham de saber binário para explicar ao computador como processar as informações. As operações que o computador executava eram representadas, também, por sequências de bits específicas. Os primeiros programadores tinham que fornecer as sequências de bits que representavam operações de leitura de dados, somas, subtrações, exibição de dados, dentre outras, bit a bit, sem erro.

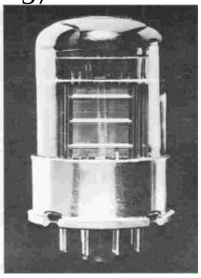
Considerando que tudo o que os computadores conseguem entender são sequências enormes de números binários, fornecer estas "instruções" pode ser muito penoso para um ser humano. Você consegue se imaginar lendo um manual de 5.000 páginas e decorando um a um centenas de códigos binários que representam as instruções do processador?

Se os programadores precisassem programar diretamente em binários, decorando sequências como 10111011101101101110110011001010 para cada instrução do processador e para cada endereço de memória a ser acessado, provavelmente não teríamos mais programadores; já estariam todos loucos.

Para facilitar as coisas, começaram a ser desenvolvidas as linguagens de programação, que diferem na sintaxe e recursos, mas têm um ponto em comum, que é a existência de um compilador. Seja programando em C, ou seja em Java, em Delphi, Kylix ou C#, dentre muitas outras linguagens, você usará um editor para escrever seu programa, respeitando as regras da linguagem escolhida e em seguida executará o compilador, que interpretará os comandos que formam o seu programa e os transformará em um arquivo binário, que contém as instruções binárias que são entendidas pelo processador.

A vantagem de usar linguagens de programação é que o desenvolvedor passa a trabalhar com instruções como "if", "else", etc. além de todas as facilidades oferecidas pela linguagem ao invés de gigantescos endereços binários. Sem dúvida muito mais simples do que trabalhar com sequências de bits.

Assim como nas línguas faladas, existem diferenças de sintaxe, gramática e existem linguagens mais simples ou mais complicadas de aprender e linguagens mais adequadas para cada tipo de tarefa a realizar. O **Assembly** foi provavelmente a primeira linguagem de programação da história, surgida na década de 50, época em que os computadores ainda usavam válvulas (uma válvula



ligada = bit 1, desligada = bit 0 → imagine a quantidade de calor gerada; veja detalhes em http://www.guiadohardware.net/termos/valvula).

A ideia do Assembly é usar um comando em substituição a cada instrução de máquina. Ela manipula diretamente as instruções e endereços de memória e, por isso, é mais trabalhosa e voltada para o desenvolvimento de aplicativos otimizados que exigem, inclusive, domínio de bases binária, octal e hexadecimal para explorar seus recursos com profundidade.

O Assembly é classificado como uma linguagem de baixo nível, o que significa que suas instruções são muito parecidas com as instruções de linguagem de máquina em binário.

Linguagens de alto nível, como Delphi, COBOL, Java, C#, Visual Basic, Smalltalk e outras, utilizam palavras e tipos de instruções mais próximas da linguagem humana.

Na verdade, as instruções que formam o Assembly são “apelidos” ou “mnemônicos” (palavras que ajudam a lembrar sua utilização) que simplesmente substituem os valores em binário correspondentes, como veremos a seguir.

INSTRUÇÃO DE MÁQUINA

Quem executa um programa é o hardware e o que ele espera encontrar é um programa em linguagem de máquina (uma sequência de instruções de máquina em código binário). Um programa em linguagem de alto nível não pode ser executado diretamente pelo *hardware*. Ele tem que ser transformado (traduzido) para linguagem de máquina por um compilador, antes de ser carregada em memória, para que o hardware possa executá-lo. A linguagem de máquina é composta de códigos binários, representando instruções, endereços e dados e está totalmente vinculada ao conjunto (*"set"*) de instruções da máquina.

Funcionalmente as operações do computador são:

- matemáticas (aritméticas, lógicas, de complemento, de deslocamento...)
- movimentação de dados (memória <--> registrador)
- entrada-saída (leitura e escrita em dispositivos externos - dispositivos de Entrada / Saída)
- controle (desvio da sequência de execução, parar, etc...)

O conjunto de instruções de alguns processadores, como por exemplo o Intel 8080 (precursor da maioria dos processadores dos PCs atuais), não possui instruções para multiplicação ou divisão; portanto, um programa em linguagem de máquina, nestes processadores, não pode fazer multiplicação ou divisão diretamente (somente através de combinação de outras instruções).

Já um programa em linguagem de nível mais alto pode implementar comandos de multiplicação (ou divisão) que combinem uma série de instruções binárias do conjunto de instruções para fazer uma multiplicação (ou divisão) através de repetidas somas (subtrações) ou por deslocamento de bits.

Exemplo de um conjunto de instruções de uma máquina hipotética:

Instrução	**Significado**	**Operação**	**Código Binário**
Load	Carregar no acumulador	ACC <-- op	0000
Store	Salvar na memória	op <-- ACC	0001
Add	Somar	ACC <-- ACC + op	0010
Sub	Subtrair	ACC <-- ACC - op	0011
Mult	Multiplicar	ACC <-- ACC * op	0100
Div	Dividir	ACC <-- ACC / op	0101
Jmp	Desviar	CI <-- op	0110
Jz	Desviar, se ACC igual zero	CI <-- op, se ACC = 0	0111
Jnz	Desviar, se ACC não zero	CI <-- op, se ACC != 0	1000
Read	Ler entrada	op <-- entrada	1001
Print	Imprimir	saida <-- op	1010
Stop	Terminar		1100

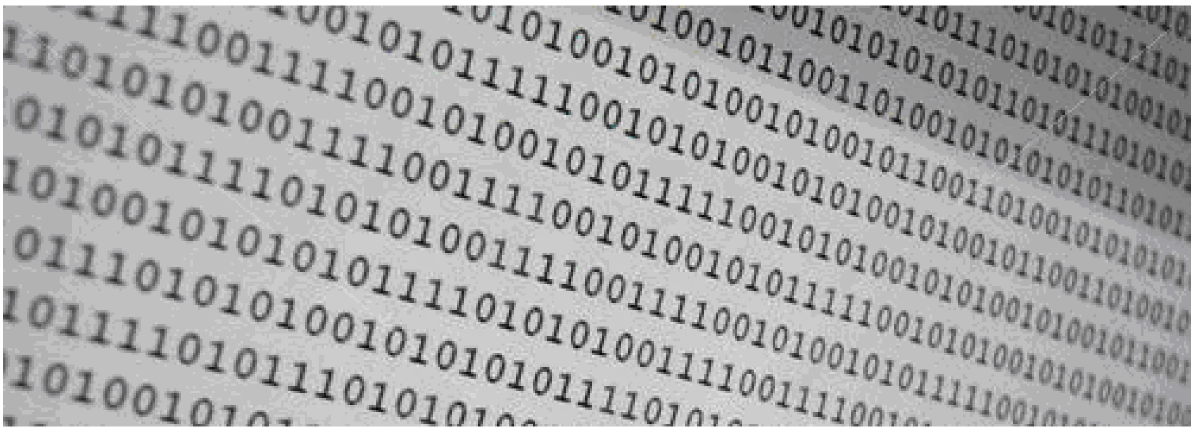
Cada uma das instruções tem um código binário associado, que é o código da operação.

Os antigos programadores, portanto, tinham que saber a combinação exata de bits desligados e ligados (0s e 1s) que representava cada instrução, para introduzí-las no computador passo a passo, para que elas fossem executadas.

Mas, atualmente, isso não é necessário. Estudaremos a seguir, portanto, como criar programas de computadores usando instruções de alto nível, sem a necessidade de usarmos os códigos binários. No entanto, é importante conhecer as técnicas de conversão de bases, bem como alguns valores binários importantes, pois eles poderão ser úteis em algumas aplicações específicas de computafores.

Fontes:

http://pt.wikipedia.org/wiki/Conversão_de_base_numérica
http://wwwusers.rdc.puc-rio.br/rmano/sn2cvb.html
http://wwwusers.rdc.puc-rio.br/rmano/ri1finst.html
http://www.inf.ufsc.br/ine5365/sistnum.html
http://pt.wikipedia.org/wiki/Sistema_binário_(matemática)
http://cti2009vc1.blogspot.com/2009/09/linguagens-de-programacao.html



www.fotosearch.com (bits)

3. Problemas Matemáticos

Vamos revisar alguns assuntos importantes da Matemática através da solução de problemas, a partir dos quais poderemos desenvolver nossa lógica de programação de maneira mais adequada.

Alguns dos problemas que estudaremos são:

- Equação do 2º Grau
- Formação e Propriedades de Triângulos
- Contagem
- Divisores de um número inteiro
- Números Primos
- MDC
- Fatoriais
- Sequências de inteiros

http://pessoal.sercomtel.com.br/matematica/fundam/eq2g/eq2g.htm

Equações do segundo grau

Introdução às equações algébricas

Equações algébricas são equações nas quais a incógnita x está sujeita a operações algébricas como: adição, subtração, multiplicação, divisão e radiciação.

Exemplos:

1. $a x + b = 0$
2. $a x^2 + bx + c = 0$
3. $a x^4 + b x^2 + c = 0$

Uma equação algébrica está em sua forma canônica, quando ela pode ser escrita como:

$$a_o x^n + a_1 x^{n-1} + \dots + a_{n-1} x^1 + a_n = 0$$

onde n é um número inteiro positivo (número natural). O maior expoente da incógnita em uma equação algébrica é denominado o grau da equação e o coeficiente do termo de mais alto grau é denominado coeficiente do termo dominante.

Exemplo: A equação $4x^2+3x+2=0$ tem o grau 2 e o coeficiente do termo dominante é 4. Neste caso, dizemos que esta é uma equação do segundo grau.

A fórmula quadrática de Sridhara (Bhaskara)

Mostraremos na sequência como o matemático Sridhara, obteve a Fórmula (conhecida como sendo) de Bhaskara, que é a fórmula geral para a resolução de equações do segundo grau. Um fato curioso é que a Fórmula de Bhaskara não foi descoberta por ele mas pelo matemático hindu

Sridhara, pelo menos um século antes da publicação de Bhaskara, fato reconhecido pelo próprio Bhaskara, embora o material construído pelo pioneiro não tenha chegado até nós.

O fundamento usado para obter esta fórmula foi buscar uma forma de reduzir a equação do segundo grau a uma do primeiro grau, através da extração de raízes quadradas de ambos os membros da mesma.

Seja a equação:

$$a x^2 + b x + c = 0$$

com a não nulo e dividindo todos os coeficientes por a, temos:

$$x^2 + (b/a) x + c/a = 0$$

Passando o termo constante para o segundo membro, teremos:

$$x^2 + (b/a) x = -c/a$$

Prosseguindo, faremos com que o lado esquerdo da equação seja um quadrado perfeito e para isto somaremos o *quadrado de b/2a* a ambos os membros da equação para obter:

$$x^2 + (b/a) x + (b/2a)^2 = -c/a + (b/2a)^2$$

Simplificando ambos os lados da equação, obteremos:

$$[x+(b/2a)]^2 = (b^2 - 4ac) / 4a^2$$

Notação: Usaremos a notação R[x] para representar a raiz quadrada de $x \geq 0$. R[5] representará a raiz quadrada de 5. Esta notação está sendo introduzida aqui para fazer com que a página seja carregada mais rapidamente, pois a linguagem HTML ainda não permite apresentar notações matemáticas na Internet de uma forma fácil.

Extraindo a raiz quadrada de cada membro da equação e lembrando que a raiz quadrada de todo número real não negativo é também não negativa, obteremos duas respostas para a nossa equação:

$$x + (b/2a) = + R[(b^2-4ac) / 4a^2]$$

ou

$$x + (b/2a) = - R[(b^2-4ac) / 4a^2]$$

que alguns, por *preguiça* ou *descuido*, escrevem:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

contendo um sinal ± que é lido como *mais ou menos*. Lembramos que este sinal ± não tem qualquer significado em Matemática.

Como estamos procurando duas raízes para a equação do segundo grau, deveremos sempre escrever:

x' = -b/2a + R[b²-4ac] /2a

ou

x" = -b/2a - R[b²-4ac] /2a

A fórmula de Bhaskara ainda pode ser escrita como:

$$x = \frac{-b \pm \sqrt{D}}{2a}$$

onde *D* (às vezes usamos a letra maiúscula "delta" do alfabeto grego) é o discriminante da equação do segundo grau, definido por:

D = b² - 4ac

Equação do segundo grau

Uma equação do segundo grau na incógnita x é da forma:

a x² + b x + c = 0

onde os números reais a, b e c são os coeficientes da equação, sendo que a deve ser diferente de zero. Essa equação é também chamada de *equação quadrática*, pois o termo de maior grau está elevado ao quadrado.

Equação Completa do segundo grau

Uma equação do segundo grau é completa, se todos os coeficientes a, b e c são diferentes de zero.

Exemplos:

1. $2x^2 + 7x + 5 = 0$
2. $3x^2 + x + 2 = 0$

Equação incompleta do segundo grau

Uma equação do segundo grau é incompleta se b=0 ou c=0 ou b=c=0. Na equação incompleta o coeficiente a é diferente de zero.

Exemplos:

1. $4x^2 + 6x = 0$
2. $3x^2 + 9 = 0$
3. $2x^2 = 0$

Resolução de equações incompletas do 2o. grau

Equações do tipo $ax^2=0$: Basta dividir toda a equação por a para obter:

$$x^2 = 0$$

significando que a equação possui duas raízes iguais a zero.

Equações do tipo $ax^2+c=0$: Novamente dividimos toda a equação por a e passamos o termo constante para o segundo membro para obter:

$$x^2 = -c/a$$

Se -c/a for negativo, não existe solução no conjunto dos números reais.

Se -c/a for positivo, a equação terá duas raízes com o mesmo valor absoluto (módulo) mas de sinais contrários.

Equações do tipo $ax^2+bx=0$: Neste caso, fatoramos a equação para obter:

$$x\ (ax + b) = 0$$

e a equação terá duas raízes:

$$x' = 0 \text{ ou } x'' = -b/a$$

Exemplos gerais

1. $4x^2=0$ tem duas raízes nulas.
2. $4x^2-8=0$ tem duas raízes: x'=R[2], x"= -R[2]
3. $4x^2+5=0$ não tem raízes reais.
4. $4x^2-12x=0$ tem duas raízes reais: x'=3, x"=0

Exercícios: Resolver as equações incompletas do segundo grau.

1. $x^2 + 6x = 0$
2. $2\ x^2 = 0$
3. $3\ x^2 + 7 = 0$
4. $2\ x^2 + 5 = 0$
5. $10\ x^2 = 0$
6. $9\ x^2 - 18 = 0$

Resolução de equações completas do 2o. grau

Como vimos, uma equação do tipo: $ax^2+bx+c=0$, é uma equação completa do segundo grau e para resolvê-la basta usar a fórmula quadrática (atribuída a Bhaskara), que pode ser escrita na forma:

$$x = \frac{-b \pm \sqrt{D}}{2a}$$

onde *D*=b²-4ac é o discriminante da equação.

Para esse discriminante *D* há três possíveis situações:

1. Se *D*<0, não há solução real, pois não existe raiz quadrada real de número negativo.
2. Se *D*=0, há duas soluções iguais:

x' = x" = -b / 2a

3. Se *D*>0, há duas soluções reais e diferentes:

x' = (-b + R[*D*])/2a
x" = (-b - R[*D*])/2a

Exemplos: Preencher a tabela com os coeficientes e o discriminante de cada equação do segundo grau, analisando os tipos de raízes da equação.

Equação	a	b	c	Delta	Tipos de raízes
$x^2-6x+8=0$	1	-6	8	4	reais e diferentes
$x^2-10x+25=0$					
$x^2+2x+7=0$					
$x^2+2x+1=0$					
$x^2+2x=0$					

O uso da fórmula de Bhaskara

Você pode realizar o Cálculo das Raízes da Equação do segundo grau com a entrada dos coeficientes a, b e c em um programa de computador, mesmo no caso em que *D* é negativo, o que força o programa a informar a existência de raízes complexas. Você pode usar a fórmula de Bhaskara mesmo para o cálculo das raízes de equações incompletas.

Mostraremos agora como usar a fórmula de Bhaskara para resolver a equação:

$$x^2 - 5x + 6 = 0$$

1. Identificar os coeficientes: a=1, b= -5, c=6
2. Escrever o discriminante D = b²-4ac.
3. Calcular D=(-5)²-4×1×6=25-24=1
4. Escrever a fórmula de Bhaskara:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

5. Substituir os valores dos coeficientes a, b e c na fórmula:

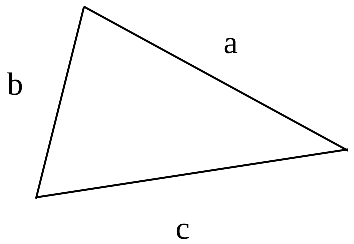
x' = (1/2)(5+R[1]) = (5+1)/2 = 3
x" = (1/2)(5-R[1]) = (5-1)/2 = 2

Exercícios

1. Calcular o discriminante de cada equação e analisar as raízes em cada caso:
   a. $x^2 + 9x + 8 = 0$
   b. $9x^2 - 24x + 16 = 0$
   c. $x^2 - 2x + 4 = 0$
   d. $3x^2 - 15x + 12 = 0$
   e. $10x^2 + 72x - 64 = 0$
2. Resolver as equações:
   a. $x^2 + 6x + 9 = 0$
   b. $3x^2 - x + 3 = 0$
   c. $2x^2 - 2x - 12 = 0$
   d. $3x^2 - 10x + 3 = 0$

Formação e Propriedades de Triângulos

http://pt.wikipedia.org/wiki/Tri%C3%A2ngulo#Tipos_de_tri.C3.A2ngulos



No plano, **triângulo** (também aceito como ***trilátero***) é a figura geométrica que ocupa o espaço interno limitado por três linhas retas que concorrem, duas a duas, em três pontos diferentes formando três lados e três ângulos internos que somam 180°.

O triângulo é o único polígono que não possui diagonais e cada um de seus ângulos externos é suplementar do ângulo interno adjacente.[2] O perímetro de um triângulo é a soma das medidas dos seus lados.

Condição de existência de um triângulo

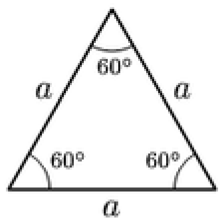
Para que se possa construir um triângulo é necessário que a medida de qualquer um dos lados seja menor que a soma das medidas dos outros dois e maior que o valor absoluto da diferença entre essas medidas.

$$|b - c| < a < b + c$$

$$a + b > c \quad e \quad b + c > a \quad e \quad a + c > b$$

Tipos de triângulos

Os triângulos mais simples são classificados de acordo com os limites das proporções relativas de seus lados e de seus ângulos internos:[1]

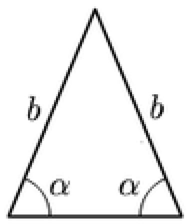


- Um **triângulo equilátero** possui todos os lados congruentes, ou seja, iguais.

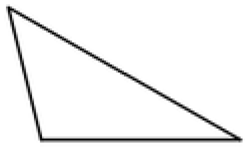
  Um triângulo equilátero é também **equiângulo**: todos os seus ângulos internos são congruentes (medem 60°), sendo, portanto, classificado como um polígono regular.

- Um **triângulo isósceles** possui pelo menos dois lados de mesma medida e dois ângulos congruentes.

Triângulo equilátero



Triângulo isósceles



Triângulo escaleno

Em relação a algoritmos, podemos desenvolver uma lógica de programa que verifique se três segmentos de reta formam um triângulo e de que tipo de triângulo se trata. Os passos básicos seriam os seguintes:

1. Solicitar ao usuário que forneça as medidas dos três segmentos de reta que supostamente formam o triângulo
2. Ler essas medidas armazenando-as em variáveis de tipo adequado
3. Testar as condições de existência de triângulo através das variáveis que armazenam as medidas
4. Se as medidas permitirem a formação de um triângulo
    a. Verificar se as medidas são todas iguais, caso em que temos um triângulo equilátero ou se há diferença de medidas e, através dessas diferenças, determinar que tipo de triângulo temos (escaleno ou isósceles).
    b. Se o triângulo não for equilátero, verificar se ele forma um triângulo retângulo. Nesse caso, o quadrado da medida do maior segmento é igual à soma do quadrado da medida de cada um dos demais lados. Mas, se não soubermos qual dos lados é o maior, basta verificarmos as três possibilidades:

$$a^2 = b^2 + c^2 \quad ou \quad b^2 = a^2 + c^2 \quad ou \quad c^2 = a^2 + b^2$$

Contagem

Contagem é uma das mais importantes operações computacionais. Ela é baseada na matemática e permite ao computador calcular quantidades de dados processados.

A contagem se baseia na busca do sucessor de um número inteiro. O sucessor de um número inteiro é dado pela soma de uma unidade ao número inteiro. Essa operação é chamada de incremento.

Se você incrementar um número inteiro sucessivas vezes, gerará a sequência de números inteiros i, i+1, i+2, i+3, ..., i+n, onde n é a quantidade de números gerados na sequência.

A variável i na sequência acima é chamada de contador. O valor inicial de um contador pode ser qualquer valor inteiro mas, em geral, se usam os valores 0 ou 1, dependendo dos eventos que se deseja contar.

Por exemplo, se o contador i tem valor inicial 0, o sucessor de i será 0 + 1 = 1. O sucessor de 1 é 1 +1 = 2. O sucessor de 2 é 2 + 1 = 3.

Podemos sistematizar a geração dos valores da sequência de inteiros da seguinte maneira:

Valor inicial i = 1

i	Sucessor	Novo i
1	i + 1 = 1 + 1	i = 2
2	i + 1 = 2 + 1	i = 3
3	i + 1 = 3 + 1	i = 4
4	i + 1 = 4 + 1	i = 5
...	i + 1 ...	...
n-1	(n-1) + 1	i = n
n	i + 1 = n + 1	I = n + 1

Na tabela ao lado, vemos a geração da sequência de valores i, onde i começa com 1 e vai sendo modificado a cada nova iteração, através do incremento de uma unidade ao seu valor.

Você pode verificar que o novo valor de i é, a cada iteração, o seu sucessor, ou seja, i passa a valer o resultado de i + 1.

Em termos de algoritmos, podemos usar o comando **i := i +1** para representar essa situação.

Como você pode deduzir, a operação de contagem é uma operação realizada dentro de um processo iterativo, ou seja, de uma repetição de operações.

Podemos, portanto, usar a contagem para várias aplicações como, por exemplo, escrever na tela do computador os valores inteiros de 1 a n, um a um, como vemos no trecho abaixo:

```
i := 1;
Enquanto i <= n faça
Inicio
...
  i := i + 1;
Fim;
```

Uma aplicação da Contagem – relação de divisores de um inteiro

Problema : Encontrar os divisores de um número natural

Vamos imaginar que nosso usuário deseja saber quais são os divisores de um número natural qualquer. Sabemos que os números naturais são os inteiros maiores que zero e que todo número natural possui pelo menos dois divisores : 1 e o próprio número.

Vamos chamar esse número de N.

Os ***possíveis*** divisores de N são todos os números naturais da sequência de 1 até N, ou seja, os valores inteiros 1 <= i <= N poderão ser divisores de N.

Nem todos esses valores, em geral, são divisores de N. Somente serão divisores de N aqueles números naturais que atendam à seguinte propriedade:

Se Resto (N / i) = 0, onde 1 <= i <= N, então i é divisor de N

Por exemplo:

Seja N igual a 18 (ou seja, nosso usuário digitou o valor 18 para saber quais são seus divisores)

Os possíveis divisores de 18 são, portanto, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17 e 18.

Os divisores ***efetivos*** de 18 são os números naturais da sequência acima descrita em que o resto da divisão de 18 pelo número resulte em zero. Portanto, temos que verificar, para cada número da sequência acima, se o resto da divisão de 18 pelo número é igual a zero. Se for, o número é divisor de 18. Se o resto da divisão não for zero, o número testado não é divisor de 18. Podemos resumir da seguinte maneira:

i	Quociente N div i	Resto da Divisão N mod i	É divisor? Resto = 0 ?	Sucessor	Novo i
1	18	0	Sim	i + 1 = 1 + 1	i = 2
2	9	0	Sim	i + 1 = 2 + 1	i = 3
3	6	0	Sim	i + 1 = 3 + 1	i = 4
4	4	2		i + 1 = 4 + 1	i = 5
5	3	3		i + 1 = 5 + 1	I = 6
6	3	0	Sim	i + 1 = 6 + 1	I = 7
7	2	4		i + 1 = 7 + 1	I = 8
8	2	2		i + 1 = 8 + 1	I = 9
9	2	0	Sim	i + 1 = 9 + 1	I = 10
10	1	8		i + 1 = 10 + 1	I = 11
11	1	7		i + 1 = 11 + 1	I = 12
12	1	6		i + 1 = 12 + 1	I = 13
13	1	5		i + 1 = 13 + 1	I = 14
14	1	4		i + 1 = 14 + 1	I = 15
15	1	3		i + 1 = 15 + 1	I = 16
16	1	2		i + 1 = 16 + 1	I = 17
17	1	1		i + 1 = 17 + 1	I = 18
18	1	0	Sim	i + 1 = 18 + 1	I = 19
19	Fim do processo repetitivo				

Acima, podemos ver que i é uma variável que foi sendo incrementada a cada iteração (ou a cada repetição). O valor de i a cada iteração foi usado como divisor para calcular o quociente e o resto da divisão (resto = N mod i). Em seguida, se o resto calculado é igual a zero, temos que i é divisor de N. Após cada verificação, incrementamos i para prepará-lo para a iteração seguinte.

Observe que, após a metade do número, só haverá um divisor, o próprio número. Isso significa que podemos modificar um pouco nossa abordagem da solução do problema e verificar apenas metade da sequência de possíveis divisores, pois sabemos que 1 e N sempre serão divisores de N.

Outra aplicação da Contagem – verificar se um inteiro é primo

Uma outra aplicação da contagem é verificar se um número inteiro é primo. Podemos usar a idéia desenvolvida no exemplo anterior para essa verificação. Mas, antes, vamos relembrar o que é um número primo.

A definição de número primo é :

[http://pt.wikipedia.org/wiki/N%C3%BAmero_primo]

Um número natural é um **número primo** quando ele tem exatamente dois divisores naturais distintos: o número um e ele mesmo[1].

Existem infinitos números primos, como demonstrado por Euclides por volta de 300 a.C..[2]

A propriedade de ser um primo é chamada "primalidade", e a palavra "primo" também é utilizada como substantivo ou adjetivo. Como "dois" é o único número primo par, o termo "primo ímpar" refere-se a todo primo maior do que dois.

Se um número inteiro tem módulo maior que um e não é primo, diz-se que é **composto**. Por convenção, os números 0, 1 e -1 não são considerados primos nem compostos.

O conceito de número primo é muito importante na teoria dos números.

[http://pt.wikipedia.org/wiki/Teoria_dos_n%C3%BAmeros]

Um dos resultados da teoria dos números é o Teorema Fundamental da Aritmética, que afirma que qualquer número natural diferente de 1 pode ser escrito de forma única (desconsiderando a ordem) como um produto de números primos (chamados **fatores primos**): este processo se chama decomposição em fatores primos (fatoração).

Assim, resumindo, podemos deduzir que:

- número N é primo quando **possui apenas dois divisores**: 1 e N.
- um número N é primo se ele não possui **nenhum divisor** entre 2 e N – 1.
- se um número N tiver qualquer divisor i, onde 2 <= i < N, então N não é primo.

As afirmações acima são equivalentes em termos de primalidade (ou não) de um número N natural qualquer. Podemos usar essas afirmações para verificar se um número N qualquer é primo.

Por exemplo, basta contarmos quantos divisores N possui entre 2 e N-1. Se essa contagem resultar em 0, o número é primo. Ou, contamos quantos divisores N possui entre 1 e N. Se essa contagem resultar em 2, o número é primo.

Os números inteiros entre 1 e N são possíveis divisores de N, como vimos no exemplo anterior. Sabemos que nem todos os valores dessa sequência serão divisores efetivos. Na verdade, no caso da solução do problema de verificar se um número é primo, o que desejamos é não encontrarmos divisores entre 2 e N-1. Mas teremos de verificar cada um dos divisores da sequência de 2 a N-1, pois não sabemos, a priori, quais são ou não divisores de N.

Podemos melhorar um pouco a nossa lógica lembrando que 2 é o único primo par e que todos os demais primos são ímpares. Assim, a tabela abaixo procura sistematizar o processo de verificação:
Seja N igual a 17 (ou seja, nosso usuário digitou o valor 17 para saber se é primo)

Os possíveis divisores de 17 são, portanto, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 e 17. Sabemos, de antemão, que 1 e N são os únicos divisores permitidos de um número primo e, portanto, não precisaremos testá-los. Também usaremos apenas potenciais divisores ímpares. Assim, nossa sequência será 3, 5, 7, 9, 11, 13 e 15.

Os divisores ***efetivos*** de 17 serão os números naturais da sequência acima descrita em que o resto da divisão de 17 pelo número resulta em zero. Portanto, temos que verificar, para cada número da sequência acima, se o resto da divisão de 17 pelo número é igual a zero. Se for, o número é divisor de 17 e contamos esse divisor. Se o resto da divisão não for zero, o número testado não é divisor de 17. Podemos resumir da seguinte maneira:

i	Quociente N div i	Resto da Divisão N mod i	É divisor? Resto = 0 ?	Quantos Divisores	Sucessor	Novo i
3	5	2	Não	0	i + 2 = 3 + 2	i = 5
5	3	2	Não	0	i + 2 = 5 + 2	i = 7
7	2	4	Não	0	i + 2 = 7 + 2	I = 9
9	1	8	Não	0	i + 2 = 9 + 2	I = 11
11	1	6	Não	0	i + 2 = 11 + 2	I = 13
13	1	4	Não	0	i + 2 = 13 + 2	I = 15
15	1	2	Não	0	i + 2 = 15 + 2	I = 17
17	Fim do processo repetitivo					

Como QuantosDivisores é igual a zero, isso significa que não achamos nenhum divisor entre 2 e N-1 e, portanto, N é primo.

Já se N fosse igual a 21, teríamos a seguinte situação:

i	Quociente N div i	Resto da Divisão N mod i	É divisor? Resto = 0 ?	Quantos Divisores	Sucessor	Novo i
3	7	0	Sim	1	i + 2 = 3 + 2	i = 5
5	4	1	Não	1	i + 2 = 5 + 2	i = 7
7	3	0	Sim	2	i + 2 = 7 + 2	I = 9
9	2	3	Não	2	i + 2 = 9 + 2	I = 11
11	Fim do processo repetitivo					

Como QuantosDivisores é igual a 2, isso significa que achamos dois divisores entre 3 e N-1 e, portanto, 21 não é primo.

Nos processos acima, levamos em consideração que o número N é ímpar. Se N fosse par e diferente de 2, ele não será primo.

Exercícios:

1. Codifique um algoritmo que exiba na tela os múltiplos de 3 entre 1 e um valor N digitado pelo usuário.

2. Crie tabelas semelhantes às dos exemplos para verificar se os valores 19, 33 e 51 são ou não primos.

3. Crie tabelas semelhantes às dos exemplos para relacionar os divisores de 14, 24, 37.

4. Codifique um algoritmo para escrever na tela os divisores de um número N digitado pelo usuário.

5. Codifique um algoritmo para escrever na tela se um número N digitado pelo usuário é ou não primo.

Fatorial - http://pt.wikipedia.org/wiki/Fatorial

Na matemática, o **fatorial** (AO 1945: factorial) de um número natural *n*, representado por *n!*, é o produto de todos os inteiros positivos menores ou iguais a *n*. A notação *n!* foi introduzida por Christian Kramp em 1808.

A função fatorial é normalmente definida por:

$$n! = \prod_{k=1}^{n} k \qquad \forall n \in \mathbb{N}$$

Por exemplo,

$$5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$$

Note que esta definição implica em particular que

$$0! = 1$$

porque o produto vazio, isto é, o produto de nenhum número é 1. Deve-se prestar atenção neste valor pois este faz com que a função recursiva

$$(n+1)! = n! \times (n+1)$$

funcione para $n = 0$.

A sequência dos fatoriais para $n = 0, 1, 2, \ldots$ começa com:

1, 1, 2, 6, 24, 120, 720, 5040, 40320, 362880, 3628800,...

Multifatoriais

Uma notação relacionada comum é o uso de múltiplos pontos de exclamação para simbolizar um **multifatorial**, o produto de inteiros em passos de dois (*n*!!), três (*n*!!!), ou mais.

n!! denota o **fatorial duplo** de *n* e é definido recursivamente por

$$n!! = \begin{cases} 1, & \text{se } n = 0 \text{ ou } n = 1; \\ n(n-2)!! & \text{se } n \geq 2. \end{cases}$$

Por exemplo, $8!! = 2 \cdot 4 \cdot 6 \cdot 8 = 384$ e $9!! = 1 \cdot 3 \cdot 5 \cdot 7 \cdot 9 = 945$. A sequência de fatoriais duplos para $n = 0, 1, 2, \ldots$ é :1, 1, 2, 3, 8, 15, 48, 105, 384, 945, 3840, ...

Hiperfatoriais

Ocasionalmente o **hiperfatorial** de *n* é considerado. É escrito como *H*(*n*) e definido por

$$H(n) = \prod_{k=1}^{n} k^k = 1^1 \cdot 2^2 \cdot 3^3 \cdots (n-1)^{n-1} \cdot n^n$$

Para *n* = 1, 2, 3, 4,... os valores de *H*(*n*) são 1, 4, 108, 27648,...

A função hiperfatorial é similar à fatorial, mas produz números maiores. A taxa de crescimento desta função, contudo, não é muito maior que um fatorial regular.

Superfatoriais

Neil Sloane e Simon Plouffe definiram o **superfatorial** em 1995 como o produto dos primeiros *n* fatoriais. Assim, o superfatorial de 4 é

$$\mathrm{sf}(4) = 1! \times 2! \times 3! \times 4! = 288$$

No geral,

$$\mathrm{sf}(n) = \prod_{k=1}^{n} k! = \prod_{k=1}^{n} k^{n-k+1} = 1^n \cdot 2^{n-1} \cdot 3^{n-2} \cdots (n-1)^2 \cdot n^1.$$

A sequência de superfatoriais começa (de *n*=0) como:

1, 1, 2, 12, 288, 34560, 24883200, ... (sequência A000178 na OEIS)

Esta idéia pode ser facilmente estendida para **superduperfatorial** como o produto dos primeiros *n* superfatoriais (iniciando com n=0), assim

1, 1, 2, 24, 6912, 238878720, 5944066965504000, ... (sequência A055462 na OEIS)

e aí em diante, recursivamente para todos os **fatoriais múltiplos**, onde o *m*-fatorial de *n* é o produto dos primeiros *n (m-1)*-fatoriais, i.e.

$$\mathrm{mf}(n,m) = \mathrm{mf}(n-1,m)\mathrm{mf}(n,m-1) = \prod_{k=1}^{n} k^{\binom{n-k+m-1}{n-k}}$$

onde $\mathrm{mf}(n,0) = n$ para $n > 0$ e $\mathrm{mf}(0,m) = 1$.

Para desenvolvermos um algoritmo que calcule o fatorial de um número N, digitado pelo usuário, podemos usar a tabela de iterações abaixo e, em seguida, raciocinar sobre seu comportamento e codificar os comandos que o reproduzam:

Vamos supor que N vale 11:

Contador	**Produto atual**	**Novo Produto**
1	1	1 * 1 = 1
2	1	1 * 2 = 2
3	2	2 * 3 = 6
4	6	4 * 6 = 24
5	24	5 * 24 = 120
6	120	6 * 120 = 720
7	720	7 * 720 = 5040
8	5040	8 * 5040 = 40320
9	40320	9 * 40320 = 362880
10	362880	10 * 362880 = 3628800
11	3628800	11 * 3628800 = 39916800

Assim, o fatorial de 11 (ou seja, 11!) vale 39916800.

Codifique um algoritmo onde o usuário digita um número inteiro e seja calculado o fatorial desse número.

Multiplicação por somas sucessivas

Uma maneira de efetuar a multiplicação é por meio de várias somas em sequência. Esse método é bastante interessante para desenvolvermos o raciocínio lógico para algoritmos, visto que nele usamos operações repetitivas e contagem.

Por exemplo:

4 * 12 = 12 + 12 + 12 + 12 = 48
7 * 3 = 3 + 3 + 3 + 3 + 3 + 3 + 3 = 21

A	**B**	**Soma**	**Contagem**
5	12	0 + 12 = 12	1
5	12	12 + 12 = 24	2
5	12	24 + 12 = 36	3
5	12	36 + 12 = 48	4
5	12	48 + 12 = 60	5

A	**B**	**Soma**	**Contagem**
3	7	0 + 7 = 7	1
3	7	7 + 7 = 14	2
3	7	14 + 7 = 21	3

Observe as semelhanças e diferenças entre as duas tabelas acima.

O que fizemos em cada iteração e quando paramos de realizá-las?

Houve valores substituidos de uma iteração para outra?
Essas observações poderão lhe ajudar a desenvolver o algoritmo desse tipo de multiplicação.

Divisão por subtrações sucessivas

Uma maneira de efetuar a divisão é por meio de várias subtrações em sequência. Esse método é bastante interessante para desenvolvermos o raciocínio lógico para algoritmos, visto que nele usamos operações repetitivas e contagem.

Imagine que nosso usuário deseja dividir o valor A pelo valor B e que, para isso, digitou esses valores. O resultado da divisão de A por B significa, na verdade, quantas vezes B "cabe" dentro de A. Assim, podemos subtrair B de A, alterando o valor de A a cada substração e contar quantas vezes foi possível realizar essa operação.

Abaixo temos vários exemplos:

A	B	Novo A	Contagem
18	3	18 – 3 = 15	1
15	3	15 – 3 = 12	2
12	3	12 – 3 = 9	3
9	3	9 – 3 = 6	4
6	3	6 – 3 = 3	5
3	3	3 – 3 = 0	6

A	B	Novo A	Contagem
32	5	32– 5 = 27	1
27	5	27 – 5 = 22	2
22	5	22 – 5 = 17	3
17	5	17 – 5 = 12	4
12	5	12 – 5 = 7	5
7	5	7 – 5 = 2	6

A	B	Novo A	Contagem
25	6	25 – 6 = 19	1
19	6	19 – 6 = 13	2
12	6	13 – 6 = 7	3
7	6	7 – 6 = 1	4

A	B	Novo A	Contagem
200	27	200 – 27 = 173	1
173	27	173 – 27 = 146	2
146	27	146 – 27 = 119	3
119	27	119 – 27 = 92	4
92	27	92 – 27 = 65	5
65	27	65 – 27 = 38	6
38	27	38 – 27 = 11	7

Observe as semelhanças e diferenças entre as quatro tabelas acima.

Qual seria o quociente e o resto de cada divisão?

O que fizemos em cada iteração e quando paramos de realizá-las?

Houve valores substituidos de uma iteração para outra?

Essas observações poderão lhe ajudar a desenvolver o algoritmo desse tipo de divisão.

Fonte: www.somatematica.com.br/emedio/**matrizes/matrizes**.php

Matrizes

Introdução

O crescente uso dos computadores tem feito com que a teoria das matrizes seja cada vez mais aplicada em áreas como Economia, Engenharia, Matemática, Física, dentre outras. Vejamos um exemplo:

A tabela a seguir representa as notas de três alunos A, B e C em uma etapa:

	Química	Inglês	Literatura	Espanhol
A	8	7	9	8
B	6	6	7	6
C	4	8	5	9

Se quisermos saber a nota do aluno **B** em Literatura, basta procurar o número que fica na segunda linha e na terceira coluna da tabela.

Vamos agora considerar uma tabela de números dispostos em linhas e colunas, como no exemplo acima, mas colocados entre parênteses ou colchetes:

$$\begin{array}{r} \text{linha} \rightarrow \\ \\ \\ \end{array} \begin{pmatrix} 8 & 7 & 9 & 8 \\ 6 & 6 & 7 & 6 \\ 4 & 8 & 5 & 9 \end{pmatrix} \text{ ou } \begin{array}{c} \begin{bmatrix} 8 & 7 & 9 & 8 \\ 6 & 6 & 7 & 6 \\ 4 & 8 & 5 & 9 \end{bmatrix} \\ \uparrow \\ \text{coluna} \end{array}$$

Em tabelas assim dispostas, os números são chamados de **elementos**. As linhas são enumeradas *de cima para baixo* e as colunas, *da esquerda para direita:*

$$\begin{array}{r} \text{1ª linha} \rightarrow \\ \text{2ª linha} \rightarrow \\ \text{3ª linha} \rightarrow \end{array} \begin{bmatrix} 1 & 4 & 7 \\ 2 & \sqrt{3} & -3 \\ 0 & 0 & 5 \end{bmatrix}$$

$$\begin{array}{ccc} \uparrow & \uparrow & \uparrow \\ \text{1ª coluna} & \text{2ª coluna} & \text{3ª coluna} \end{array}$$

A quantidade de de linhas e a quantidade de colunas definem a Ordem de uma matriz.

Tabelas com **m** linhas e **n** colunas (**m** e **n** números naturais diferentes de 0) são denominadas matrizes m x n. Na tabela anterior temos, portanto, uma matriz 3 x 3.

Veja mais alguns exemplos:

- $\begin{bmatrix} 2 & 3 & -1 \\ 30 & -3 & 17 \end{bmatrix}$ é uma matriz do tipo 2 x 3 ou de Ordem 2 x 3
- $\begin{bmatrix} 2 & -5 \\ \frac{1}{2} & \frac{1}{3} \end{bmatrix}$ é uma matriz do tipo 2 x 2 ou de Ordem 2 x 2

Uma matriz pode ser vista como um arranjo de células ou posições dispostas em forma tabular (forma de tabela). Cada célula é identificada por um par de números representando o número da linha e o número da coluna em cuja intersecção o elemento se encontra. Por exemplo, na matriz anterior, de ordem 2x2, há 4 células. O valor 2 está na célula cuja linha é 1 e cuja coluna é 2. Assim sendo, essa célula tem as coordenadas [1,1]. Já ½ está na linha 2 e coluna 1, sendo que essa célula tem a coordenada [2,1].

Essas coordenadas são, também, chamadas de índice de linha e índice de coluna da matriz.

Notação geral

Costuma-se representar as matrizes por *letras maiúsculas* e seus elementos por *letras minúsculas*, acompanhadas pelos *dois índices* que indicam, respectivamente, a linha e a coluna que o elemento ocupa.

Assim, uma matriz **A** do tipo m x n é representada por:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & a_{n3} & \cdots & a_{mn} \end{bmatrix}$$

ou, abreviadamente, $A = [a_{ij}]_{m \times n}$, em que **i** e **j** representam, respectivamente, a linha e a coluna que o elemento ocupa. Por exemplo, na matriz anterior, $\mathbf{a_{23}}$ é o elemento da 2ª linha e da 3ª coluna.

Na matriz , $A = \begin{bmatrix} 2 & -1 & 5 \\ 4 & \frac{1}{2} & \sqrt{2} \\ 0 & 1 & -2 \end{bmatrix}$ temos: $\begin{cases} a_{11} = 2, a_{12} = -1 \text{ e } a_{13} = 5 \\ a_{21} = 4, a_{22} = \frac{1}{2} \text{ e } a_{23} = \sqrt{2} \\ a_{31} = 0, a_{32} = 1 \text{ e } a_{33} = -2 \end{cases}$

Ou na matriz B = [-1 0 2 5], temos: $a_{11} = -1$, $a_{12} = 0$, $a_{13} = 2$ e $a_{14} = 5$.

Denominações especiais

Algumas matrizes, por suas características, recebem denominações especiais.

- **Matriz linha**: matriz do tipo 1 x n, ou seja, com uma única linha. Por exemplo, a matriz A =[4 7 -3 1], do tipo 1 x 4.

- **Matriz coluna**: matriz do tipo m x 1, ou seja, com uma única coluna. Por exemplo, do tipo 3 x 1 $B = \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix}$

- **Matriz quadrada:** matriz do tipo n x n, ou seja, com o mesmo número de linhas e colunas; dizemos que a matriz é de ordem **n**. Por exemplo, a matriz $C = \begin{bmatrix} 2 & 7 \\ 4 & 1 \end{bmatrix}$ é do tipo 2 x 2, isto é, quadrada de ordem 2.

Numa matriz quadrada definimos a **diagonal principal** e a **diagonal secundária**. A principal é formada pelos elementos $\mathbf{a_{ij}}$ tais que i = j. Na secundária, temos i + j = n + 1.

Veja:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ . & . & . & \dots & \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix}$$

diagonal principal i = j

diagonal secundária i + j = n + 1

Observe a matriz a seguir:

ordem da matriz

$$A_3 = \begin{bmatrix} -1 & 2 & -5 \\ 3 & 0 & -3 \\ 5 & 7 & -6 \end{bmatrix}$$

diagonal principal

diagonal secundária

a_{11} = -1 é elemento da diagonal principal, pois i = j = 1
a_{31}= 5 é elemento da diagonal secundária, pois i + j = n + 1 (3 + 1 = 3 + 1)

- **Matriz nula**: matriz em que todos os elementos são nulos; é representada por $0_{m \times n}$.

Por exemplo, $0_{2x3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

- **Matriz diagonal**: matriz quadrada em que todos os elementos que não estão na diagonal principal são nulos. Por exemplo:

$a)\ A_{2x2} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$ $\qquad$ $b)\ B_{3x3} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 7 \end{bmatrix}$

- **Matriz identidade**: matriz quadrada em que todos os elementos da diagonal principal são iguais a 1 e os demais são nulos; é representada por $\mathbf{I_n}$**,** sendo **n** a ordem da matriz. Por exemplo:

$a)\ I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $\qquad$ $b)\ I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Assim, para uma matriz identidade $I_n = [a_{ij}],\ a_{ij} = \begin{cases} 1, se\ i = j \\ 0, se\ i \neq j \end{cases}$.

- **Matriz transposta**: matriz A^t obtida a partir da matriz A trocando-se ordenadamente as linhas por colunas ou as colunas por linhas. Por exemplo:

$$Se\ A = \begin{bmatrix} 2 & 3 & 0 \\ -1 & -2 & 1 \end{bmatrix}, então\ A^t = \begin{bmatrix} 2 & -1 \\ 3 & -2 \\ 0 & 1 \end{bmatrix}$$

Desse modo, se a matriz **A** é do tipo m x n, $\mathbf{A^t}$ é do tipo n x m.

Note que a 1ª linha de **A** corresponde à 1ª coluna de $\mathbf{A^t}$ e a 2ª linha de A corresponde à 2ª coluna de $\mathbf{A^t}$.

- **Matriz simétrica**: matriz quadrada de ordem n tal que $\mathbf{A = A^t}$ **.** Por exemplo,

$$A = \begin{bmatrix} 3 & 5 & 6 \\ 5 & 2 & 4 \\ 6 & 4 & 8 \end{bmatrix}$$

é simétrica, pois $a_{12} = a_{21} = 5$, $a_{13} = a_{31} = 6$, $a_{23} = a_{32} = 4$, ou seja, temos sempre $a_{ij} = a_{ij}$.

- **Matriz oposta**: matriz **-A** obtida a partir de **A** trocando-se o sinal de todos os
- elementos de **A**. Por exemplo, . $Se\ A = \begin{bmatrix} 3 & 0 \\ 4 & -1 \end{bmatrix}, então\ -A = \begin{bmatrix} -3 & 0 \\ -4 & 1 \end{bmatrix}$

Igualdade de matrizes

Duas matrizes, A e B, do mesmo tipo m x n, são iguais se, e somente se, todos os elementos que ocupam a mesma posição são iguais:

$$A = B \Leftrightarrow a_{ij} = b_{ij}\ para\ \text{todo } 1 \leq i \leq m \text{ e todo } 1 \leq j \leq n$$

$$Se\ A = \begin{bmatrix} 2 & 0 \\ -1 & b \end{bmatrix},\ B = \begin{bmatrix} 2 & c \\ -1 & 3 \end{bmatrix} \text{ e } A = B, \text{ então } c = 0 \text{ e } b = 3$$

.

Operações envolvendo matrizes

Adição

Dadas as matrizes $A = [a_{ij}]_{mxn}$ e $B = [b_{ij}]_{mxn}$, chamamos de soma dessas matrizes a matriz $C = [C_{ij}]_{mxn}$, tal que $C_{ij} = a_{ij} + b_{ij}$, para todo $1 \leq i \leq m \text{ e todo } 1 \leq j \leq n$:

$$\boxed{A + B = C}$$

Exemplos:

- $\begin{bmatrix} 1 & 4 \\ 0 & 7 \end{bmatrix} + \begin{bmatrix} 2 & -1 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 1+2 & 4+(-1) \\ 0+0 & 7+2 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ 0 & 9 \end{bmatrix}$

- $\begin{bmatrix} 2 & 3 & 0 \\ 0 & 1 & -1 \end{bmatrix} + \begin{bmatrix} 3 & 1 & 1 \\ 1 & -1 & 2 \end{bmatrix} = \begin{bmatrix} 2+3 & 3+1 & 0+1 \\ 0+1 & 1+(-1) & -1+2 \end{bmatrix} = \begin{bmatrix} 5 & 4 & 1 \\ 1 & 0 & 1 \end{bmatrix}$

Observação: A + B existe se, e somente se, **A** e **B** forem do mesmo tipo, ou seja, se tiverem a mesma ordem.

Propriedades
Sendo **A**, **B** e **C** matrizes do mesmo tipo (m x n), temos as seguintes propriedades para a adição:
a) comutativa: A + B = B + A
b) associativa: (A + B) + C = A + (B + C)
c) elemento neutro: A + 0 = 0 + A = A, sendo 0 a matriz nula m x n
d) elemento oposto: A + (- A) = (-A) + A = 0

Subtração

Dadas as matrizes $A = [a_{ij}]_{m \times n}$ e $B = [b_{ij}]_{m \times n}$, chamamos de diferença entre essas matrizes a soma de **A** com a matriz oposta de **B**:

$$\boxed{A - B = A + (-B)}$$

Observe:

$$\begin{bmatrix} 3 & 0 \\ 4 & -7 \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 0 & -2 \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ 4 & -7 \end{bmatrix} + \underbrace{\begin{bmatrix} -1 & -2 \\ 0 & 2 \end{bmatrix}}_{B} = \begin{bmatrix} 3+(-1) & 0+(-2) \\ 4+0 & -7+2 \end{bmatrix} = \begin{bmatrix} 2 & -2 \\ 4 & -5 \end{bmatrix}$$

Multiplicação de um número real por uma matriz (multiplicação por escalar)

Dados um número real **x** e uma matriz **A** do tipo m x n, o produto de **x** por **A** é uma matriz **B** do tipo m x n obtida pela multiplicação de cada elemento de **A** por **x**, ou seja, $b_{ij} = xa_{ij}$:

$$\boxed{B = x.A}$$

Observe o seguinte exemplo:

$$3.\begin{bmatrix} 2 & 7 \\ -1 & 0 \end{bmatrix} = \begin{bmatrix} 3.2 & 3.7 \\ 3.(-1) & 3.0 \end{bmatrix} = \begin{bmatrix} 6 & 21 \\ -3 & 0 \end{bmatrix}$$

Propriedades

Sendo **A** e **B** matrizes do mesmo tipo (m x n) e **x** e **y** números reais quaisquer, valem as seguintes propriedades:
a) associativa: x . (yA) = (xy) . A
b) distributiva de um número real em relação à adição de matrizes: x . (A + B) = xA + xB
c) distributiva de uma matriz em relação à adição de dois números reais: (x + y) . A = xA + yA
d) elemento neutro : xA = A, para x=1, ou seja, A=A

Multiplicação de matrizes

O produto de uma matriz por outra não é determinado por meio do produto dos seus respectivos elementos.

Assim, o produto das matrizes $A = (a_{ij})_{m \times p}$ e $B = (b_{ij})_{p \times n}$ é a matriz $C = (c_{ij})_{m \times n}$ em que cada elemento $\mathbf{c_{ij}}$ é obtido por meio da soma dos produtos dos elementos correspondentes da i-ésima linha de A pelos elementos da j-ésima coluna **B**.

Vamos multiplicar a matriz $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ e $B = \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix}$ para entender como se obtém cada $\mathbf{C_{ij}}$:

- 1ª linha e 1ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} \overset{c_{11}}{\boxed{1.(-1)+2.4}} & \\ & \end{bmatrix}$$

- 1ª linha e 2ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} 1.(-1)+2.4 & \overset{c_{12}}{\boxed{1.3+2.2}} \\ & \end{bmatrix}$$

- 2ª linha e 1ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} 1.(-1)+2.4 & 1.3+2.2 \\ \underset{c_{21}}{\boxed{3.(-1)+4.4}} & \end{bmatrix}$$

- 2ª linha e 2ª coluna

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} = \begin{bmatrix} 1.(-1)+2.4 & 1.3+2.2 \\ 3.(-1)+4.4 & \underset{c_{22}}{\boxed{3.3+4.2}} \end{bmatrix}$$

Assim, $A.B = \begin{bmatrix} 7 & 7 \\ 13 & 17 \end{bmatrix}$.

Observe que:

$$A.B = \begin{bmatrix} -1 & 3 \\ 4 & 2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} \boxed{(-1).1+3.3} & \boxed{(-1).2+3.4} \\ \boxed{4.1+2.3} & \boxed{4.2+2.4} \end{bmatrix} = \begin{bmatrix} 8 & 10 \\ 10 & 16 \end{bmatrix}$$

Portanto, A x B ≠ B x A, ou seja, para a multiplicação de matrizes não vale a propriedade comutativa.

Vejamos outro exemplo com as matrizes $A = \begin{bmatrix} 2 & 3 \\ 0 & 1 \\ -1 & 4 \end{bmatrix}$ e $B = \begin{bmatrix} 1 & 2 & 3 \\ -2 & 0 & 4 \end{bmatrix}$:

$$A.B = \begin{bmatrix} 2 & 3 \\ 0 & 1 \\ -1 & 4 \end{bmatrix} . \begin{bmatrix} 1 & 2 & 3 \\ -2 & 0 & 4 \end{bmatrix} = \begin{bmatrix} 2.1+3(-2) & 2.2+3.0 & 2.3+3.4 \\ 0.1+1(-2) & 0.2+1.0 & 0.3+1.4 \\ -1.1+4(-2) & -1.2+4.0 & -1.3+4.4 \end{bmatrix} = \begin{bmatrix} -4 & 4 & 18 \\ -2 & 0 & 4 \\ -9 & -2 & 13 \end{bmatrix}$$

$$B.A = \begin{bmatrix} 1 & 2 & 3 \\ -2 & 0 & 4 \end{bmatrix} . \begin{bmatrix} 2 & 3 \\ 0 & 1 \\ -1 & 4 \end{bmatrix} = \begin{bmatrix} 1.2+2.0+3(-1) & 1.3+2.1+3.4 \\ -2.2+0.0+4(-1) & -2.3+0.1+4.4 \end{bmatrix} = \begin{bmatrix} -1 & 17 \\ -8 & 10 \end{bmatrix}$$

Da definição, temos que a matriz produto A . B só existe se o número de colunas de **A** for igual ao número de linhas de **B**:

$$A_{m \times p} \cdot B_{p \times n} = (A.B)_{m \times n}$$

A matriz produto terá o número de linhas de **A (m)** e o número de colunas de **B(n)**:

- Se $A_{3 \times 2}$ e $B_{2 \times 5}$, então $(A.B)_{3 \times 5}$
- Se $A_{4 \times 1}$ e $B_{2 \times 3}$, então não existe o produto
- Se $A_{4 \times 2}$ e $B_{2 \times 1}$, então $(A.B)_{4 \times 1}$

Propriedades

Verificadas as condições de existência para a multiplicação de matrizes, valem as seguintes propriedades:

a) associativa: (A . B) . C = A . (B . C)
b) distributiva em relação à adição: A . (B + C) = A . B + A . C ou (A + B) . C = A . C + B . C
c) elemento neutro: $A . I_n = I_n . A = A$, sendo $\mathbf{I_n}$ a matriz identidade de ordem **n**

Vimos que a propriedade comutativa, geralmente, não vale para a multiplicação de matrizes. Não vale também o anulamento do produto, ou seja: sendo $0_{m \times n}$ uma matriz nula, $A.B = 0_{m \times n}$ não implica, necessariamente, que $A = 0_{m \times n}$ ou $B = 0_{m \times n}$.

Matriz inversa

Dada uma matriz **A**, quadrada, de ordem **n**, se existir uma matriz **A'**, de mesma ordem, tal que $A . A' = A' . A = I_n$, então **A'** é matriz inversa de **A** . Representamos a matriz inversa por $\mathbf{A^{-1}}$.

Vídeos:
Conceitos iniciais:

1 - http://www.youtube.com/watch?v=sw18GQESKpA
2 - http://www.youtube.com/watch?v=AdwyrhQ5GfA
3 - http://www.youtube.com/watch?v=V2LRnz54-dQ
4 - http://www.youtube.com/watch?v=nRD2oTiVhx8
5 - http://www.youtube.com/watch?v=Z7TLOzjL5Xo

Determinantes:
1- http://www.youtube.com/watch?v=SUbr6zypkLA

2 - http://www.youtube.com/watch?v=hjAn3Uiqaa4
3 - http://www.youtube.com/watch?v=LEl1Tu8ZimI
4 - http://www.youtube.com/watch?v=0TiAF4qPe-4
5 - http://www.youtube.com/watch?v=UMZHreMc7wo
6 - http://www.youtube.com/watch?v=HyT-54ScyFM
7 - http://www.youtube.com/watch?v=nVxhjL9NjC0
8 - http://www.youtube.com/watch?v=PTvXc41g8qA
9 - http://www.youtube.com/watch?v=oNF7UzTSv7w
10 - http://www.youtube.com/watch?v=qTnhWkEsZ6g
11 - http://www.youtube.com/watch?v=oehKFf89V0g
12 - http://www.youtube.com/watch?v=pQGx2Zg1fFs
13 - http://www.youtube.com/watch?v=kXsVhmKDc7Q
14 - http://www.youtube.com/watch?v=coDQ_No0984

Matriz Inversa:
1 - http://www.youtube.com/watch?v=SI8e3T7icX8
2 - http://www.youtube.com/watch?v=EFn8guBXqCk
3 - http://www.youtube.com/watch?v=QbwoVGijUBY
4 - http://www.youtube.com/watch?v=WgDzIkTXXCU
5 - http://www.youtube.com/watch?v=owVlYDaUHi4
6 - http://www.youtube.com/watch?v=Iot7K8dUZw8
7 - http://www.youtube.com/watch?v=rph-2sngHBE
8 - http://www.youtube.com/watch?v=7QgEtJsQo8I

Sistemas lineares:
1 - http://www.youtube.com/watch?v=HRrUF3eBFXs
2 - http://www.youtube.com/watch?v=A2JsWo_2CP8
3 - http://www.youtube.com/watch?v=VejbFBkN-4g
4 - http://www.youtube.com/watch?v=TQjInXLOj4g
5 - http://www.youtube.com/watch?v=CYIZqz0wd3U
6 - http://www.youtube.com/watch?v=EdRtsiRNLs0
7 - http://www.youtube.com/watch?v=JUFWBIq7KeE
8 - http://www.youtube.com/watch?v=kDfwjNj1cEA
9 - http://www.youtube.com/watch?v=SKcsStUST94
10 - http://www.youtube.com/watch?v=zKxtq5CQXks
11 - http://www.youtube.com/watch?v=hSS7znDKr7g
12 - http://www.youtube.com/watch?v=gt5w05x_6ro
13 - http://www.youtube.com/watch?v=siI_8JcpL20
14 - http://www.youtube.com/watch?v=A3A9Myr8zNM
15 - http://www.youtube.com/watch?v=Lq7iEDhk4XM
16 - http://www.youtube.com/watch?v=tOnSHR8CNvo
17 - http://www.youtube.com/watch?v=E-GZ0_nGBRU
18 - http://www.youtube.com/watch?v=KwQ8guhcnVY

Exercícios

http://www.ime.unicamp.br/~ra137920/exerciciosmatrizes2010.pdf

Exercícios

1. Escreva na forma de tabela as matrizes:

   a. $A = (a_{ij})2x3$, $a_{ij}=i^2 - j$;

   b. $B = (b_{ij})2x2$, $b_{ij} = 4$, se $i \geq j$ e $b_{ij} = 2.j-i$, se $i < j$;

   c. $C = (c_{ij})1x5$, $c_{ij}=i+j$, se $i=j$; $c_{ij}=i^2+j^2$, se $i>j$ e $c_{ij}=(i+j)^2$, se $i<j$;

   d. $D = (d_{ij})5x2$, $d_{ij}=2.i+3.j$, se $i=j$; $d_{ij}=2.i-3.j$, se $i>j$ e $d_{ij}=i-j^3$, se $i<j$.

2. Dada a matriz $A=(a_{ij})_{3x4}$, $a_{ij}= \sqrt{i} +2$, obtenha o elemento a'_{41} da matriz A', transposta de A.

3. Obtenha x, y e z na igualdade entre as matrizes:

a) $\begin{array}{|c|c|}\hline x-1 & 3z \\ \hline x+y & x-y \\ \hline\end{array} = \begin{array}{|c|c|}\hline 5 & x \\ \hline 8 & 4 \\ \hline\end{array}$

b) $\begin{array}{|c|c|}\hline x & z \\ \hline x+y & x-y \\ \hline\end{array} = \begin{array}{|c|c|}\hline 6 & 3x \\ \hline 8 & 4 \\ \hline\end{array}$

4. Dadas as matrizes:

$A = \begin{array}{|c|c|}\hline 5 & 0 \\ \hline 3 & 2 \\ \hline\end{array}$, $B = \begin{array}{|c|c|}\hline 4 & 2 \\ \hline -2 & 6 \\ \hline\end{array}$ e $C = \begin{array}{|c|c|}\hline -1 & 1 \\ \hline 0 & 2 \\ \hline\end{array}$, obtenha:

   a. A+B-C';

   b. –A+B/2;

   c. 2A-3B-(-C); → 2A-3B+C

   d. (A-4B)+C;

   e. (A-C)'.

5. Dadas as matrizes A e B do exercício 4, obtenha:

   a. a matriz X tal que: 4X-A+B=0, em que 0 é a matriz nula;

   b. o produto da matriz A pela matriz B;

   c. acrescente à matriz A uma linha (3ª): [1 -1] e obtenha, novamente, A.B e, posteriormente, B.A.

6. Dadas as matrizes:

$A = \begin{array}{|c|c|}\hline 1 & 2 \\ \hline 3 & 0 \\ \hline\end{array}$, $B = \begin{array}{|c|}\hline 8 \\ \hline 6 \\ \hline\end{array}$ e $C = \begin{array}{|c|c|}\hline -1 & 1 \\ \hline 0 & 2 \\ \hline\end{array}$, obter:

   a. a matriz X tal que A.X=B;

   b. a matriz Z tal que A.B + Z=C.

7. Considere a matriz C=cij de ordem 3, com cij= i / j, se $i \neq j$ e $i^j + 2j$, se i=j. Então, determine a soma dos elementos da diagonal secundária da matriz C.

8. Uma matriz quadrada A diz-se simétrica se A=A'. Assim, se a matriz

A =

2	-1	2y
x	0	-1
4	2z	3

é simétrica,então qual é o valor de x – (y+z)?

A' =

2	x	4
-1	0	2z
2y	-1	3

9. Considere três lojas L_1, L_2 e L_3 e três tipos de produtos p_1, p_2 e p_3. A matriz A descreve a quantidade de cada produto vendido em cada loja numa semana.

A =

30	19	20
15	10	8
12	16	11

Cada elemento a_{ij} da matriz indica a quantidade do produto p_i vendido pela loja L_j (i,j=1, 2, 3). Dessa forma, pede-se:

a. Quantas unidades de p3 a loja L2 vendeu nessa semana?

b. Qual o total vendido de p1 nas três lojas?

c. Seja B=[2 1 4] uma matriz tal que o elemento b1k representa o preço, em reais, do produto pk (k=1, 2, 3). Calcule B.A.

d. Observando a resposta do item c), determine o total arrecadado em cada loja, em reais, com as vendas dos três produtos.