**EXERCÍCIOS DE ALGORITMO  
QUE SERÃO USADOS NO 2º SEM/2012**

**Lote 1.1**

**1.Coletar o valor do lado de um quadrado, calcular sua área e apresentar o resultado.**

area\_quad

FIM

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

nnnmmmmmmmmmmmmmmmmmmm

area

area=Lado\*Lado

Lado, area: num

Lado

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lado | 1 | 2 | 3 | 4 |
| Area | 1 | 4 | 8 | 16 |

Massa de Dados

1

2

3

4

Pseudo-Código

Algoritmo area\_quad

Declarar.

Lado, area : num;

Início.

Ler(Lado);

area=Lado\*Lado;

Mostra (area)

Fim

**2.Receba o salário de um funcionário e mostre o novo salário com reajuste de 15%.**

Fluxograma

Salario\_reaj

FIM

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

nnnmmmmmmmmmmmmmmmmmmm

x1

x1=x+((x/100)\*15)

x,x1 :num

x

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 1 | 2 | 3 | 4 |
| x1 | 1,15 | 2,3 | 3,45 | 4,6 |

Massa de Dados

1

2

3

4

Pseudo-Código

Algoritmo Salario\_reaj

Declarar.

x,x1: num;

Início.

Ler(x);

x1=x+((x/100)\*15);

Mostra (x1)

Fim.

**3.Receba a base e a altura de um triângulo. Calcule e mostre a sua área.**

Fluxograma

area\_tring

FIM

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

nnnmmmmmmmmmmmmmmmmmmm

area

area=(b\*h)/2

b,h,area :num

b,h

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| B | 1 | 1 | 2 | 2 |
| H | 1 | 2 | 3 | 4 |
| Área | 0,5 | 1 | 3 | 4 |

Massa de Dados

B= 1 1 2 2

H= 1 2 3 4

Pseudo-Código

Algoritmo Area\_triang

Declarar.

b,h,area: num;

Início.

Ler(b);

Ler(h);

area=(b\*h)/2;

Mostra (area)

Fim.

**4.Receba a temperatura em graus Celsius. Calcule e mostre a sua temperatura convertida em fahrenheit (F = (9\*C+160)/5).**

Temp

FIM

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

nnnmmmmmmmmmmmmmmmmmmm

F

F = (9\*C+160)/5.

F,C: real

C

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C | 1 | 2 | 3 | 4 |
| F | 33,8 | 35,6 | 37,4 | 39,2 |

Massa de Dados

1

2

3

4

Pseudo-Código

Algoritmo Temp

Declarar.

C,F : real;

Início.

Ler(C);

F = (9\*C+160)/5;

Mostra (F)

Fim

**5.Receba os coeficientes A, B e C de uma equação do 2º grau (AX²+BX+C=0). Calcule e mostre as raízes reais (considerar que a equação possuem2 raízes).**

grau

A,B,C,R1,R2: int

A, B,C C

R1=(-B+raiz((B\*B)-4\*A\*C))/2\*A

R2=(-B-raiz((B\*B)-4\*A\*C))/2\*A

FIM

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

nnnmmmmmmmmmmmmmmmmmmm

R1,R2

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | 1 | 1 | 10 | 1 |
| B | -9 | 2 | 1 | -9 |
| C | 14 | 0 | 0 | 20 |
| R1 | 7 | 0 | 0 | 5 |
| R2 | 2 | -1 | -10 | 4 |

Massa de Dados

A=1, 1, 10, 1;

B=-9, 2, 1, -9;

C=14, 0, 0, 20;

Pseudo-Código

Algoritmo grau

Declarar.

A,B,C,r1,r2 : real;

Início.

Ler(A);

Ler(B);

Ler(C);

r1=(-B+raiz((B\*B)-4\*A\*C))/2\*A;

r2=(-B-raiz((B\*B)-4\*A\*C))/2\*A;

Mostra (r1,r2);

Fim

**6.Receba os valores em x e y. Efetua a troca de seus valores e mostre seus conteúdos.**

Pseudo-Código

Algoritmo

Declarar.

X,Y,X1,Y1 : num;

Início.

Ler(X);

Ler(Y);

X1=Y;

Y1=X;

X=X1;

Y=Y1;

Mostra (X,Y);

Fim

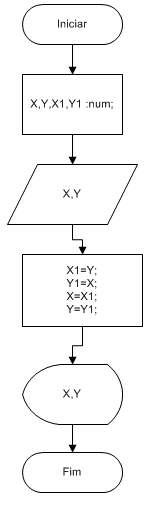
Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 1 | 2 | 1 | 2 |
| Y | 2 | 1 | 2 | 1 |
| X1 | 2 | 1 | 2 | 1 |
| Y1 | 1 | 2 | 1 | 2 |

Massa de Dados

X=1,

Y=2.



**7.Receba os valores do comprimento, largura e altura de um paralelepípedo. Calcule e mostre seu volume.**

Pseudo-Código

Algoritmo grau

Declarar.

c,l,h,v : num;

Início.

Ler(c);

Ler(l);

Ler(h);

v=c\*l\*h;

Mostra (v);

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C | 1 | 2 | 3 | 4 |
| L | 1 | 2 | 3 | 4 |
| H | 1 | 2 | 3 | 4 |
| V | 1 | 8 | 27 | 64 |

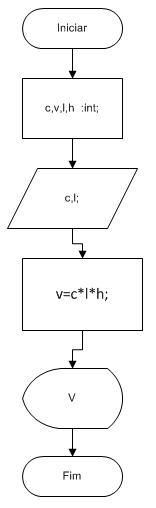
Massa de Dados

1

2

3

4



**8.Receba o valor de um depósito em poupança. Calcule e mostre o valor após 1 mês de aplicação sabendo que rende 1,3% a. m.**

Pseudo-Código

Algoritmo poupanca

Declarar.

poup: num;

Início.

Ler (poup);

poup=poup+((poup/100)\*1,3);

Mostra (poup);

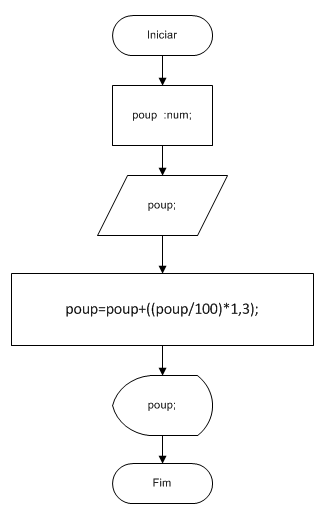
Fim

Teste de Mesa

|  |  |  |
| --- | --- | --- |
| Poup | 10 | 10,13 |

Massa de Dados

10



**9. Receba os 2 números inteiros. Calcule e mostre a soma dos quadrados.**

Pseudo-Código

Algoritmo soma\_qradrados

Declarar.

X,Y,R : int;

Início.

Ler(X);

Ler(Y);

X=X\*X;

Y=Y\*Y;

R=X+Y;

Mostra (R);

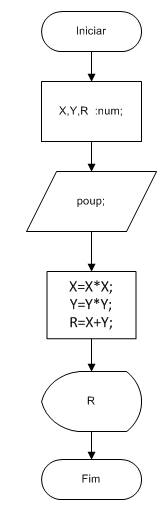
Fim

Teste de Mesa

|  |  |  |  |
| --- | --- | --- | --- |
| X | 1 | 2 | 3 |
| Y | 1 | 2 | 3 |
| R | 1 | 8 | 18 |

Massa de Dados

1, 2, 3;



**10.Receba 2 números reais. Calcule e mostre a diferença dos custos desses valores.**

Pseudo-Código

Algoritmo diferença\_real

Declarar.

X,Y,R : num;

Início.

Ler(X);

Ler(Y);

Se (X>Y) então

R=X-Y;

Se (X==Y) então

R=0;

Se(Y<X) então

R=Y-X;

Mostra (R);

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 10 | 2 | 3 | 1 |
| Y | 5 | 2 | 3 | 8 |
| R | 5 | 0 | 0 | 7 |

Massa de Dados

X=10, 2, 3, 1;

Y= 5, 2, 3, 8;

11. Receba o raio de uma circunferência. Calcule e mostre o comprimento da circunferência.

Pseudo-Código

Algoritmo circunferencia

Declarar.

X: num;

Início.

Ler(X);

X=3,14\*2\*X;

Mostra (X);

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 1 | 6,28 | 39,43 | 247,62 |

Massa de Dados

X=1;

**12.Receba o ano de nascimento e o ano atual. Calcule e mostre a sua idade e quantos anos terá daqui a 17 anos**.

Pseudo-Código

Algoritmo idade\_17

Declarar.

Ano\_nasc, ano\_atual, idade,idade\_fut: num;

Início.

Ler(Ano\_nasc);

Ler(ano\_atual);

idade=ano\_atual-Ano\_nasc;

idade\_fut=idade+17;

Mostra (idade\_fut);

Fim

Teste de Mesa

|  |  |
| --- | --- |
| Ano\_nasc | 1995 |
| ano\_atual | 2013 |
| idade | 18 |
| idade\_fut | 35 |

Massa de Dados

Ano\_nasc =1995

ano\_atual=2013

**13.Receba a quantidade de alimento em quilos. Calcule e mostre quantos dias durará esse alimento sabendo que a pessoa consome 50g ao dia.**

Pseudo-Código

Algoritmo alimentos

Declarar.

x,: num;

Início.

Ler(x);

x=x/50.

Mostra (x);

Fim

Teste de Mesa

|  |  |  |
| --- | --- | --- |
| X | 500 | 10 |

Massa de Dados

x=500;

**14. Receba 2 ângulos de um triângulo. Calcule e mostre o valor do 3º ângulo.**

Pseudo-Código

Algoritmo angu\_triang

Declarar.

ang1, ang2,ang3: num;

Início.

Ler(ang1);

Ler(ang2);

ang3=180-ang1-ang2;

Mostra (ang3);

Fim

Teste de Mesa

|  |  |
| --- | --- |
| ang1 | 90 |
| ang2 | 45 |
| ang3 | 45 |

Massa de Dados

ang1=90

ang2=45

**15. Receba os valores de 2 catetos de um triângulo retângulo. Calcule e mostre a hipotenusa.**

Pseudo-Código

Algoritmo catetos

Declarar.

cat1, cat2,h: num;

Início.

Ler(cat1);

Ler(cat2);

h=raiz((cat1\*cat1)+(cat2\*cat2);

Mostra (h);

Fim

Teste de Mesa

|  |  |
| --- | --- |
| cat1 | 4 |
| cat2 | 9 |
| h | 9,84 |

Massa de Dados

cat1=4

cat2=9

**16. Receba a quantidade de horas trabalhadas, o valor por hora, o percentual de desconto e o número de descendentes. Calcule o salário que serão as horas trabalhadas x o valor por hora. Calcule o salário líquido (= Salário Bruto – desconto). A cada dependente será acrescido R$ 100 no Salário Líquido. Exiba o salário a receber.**

Pseudo-Código

Algoritmo salario\_liq

Declarar.

ht, vh, desc, dep, salb, sal: num;

Início.

Ler(ht);

Ler(vh);

Ler(desc);

Ler(dep);

salb=ht\*vh;

sal=salb-((salb/100)\*desc)+(100\*dep);

Mostra (sal);

Fim

Teste de Mesa

|  |  |
| --- | --- |
| ht | 120 |
| vh | 50 |
| desc | 10 |
| dep | 5 |
| salb | 6000 |
| sal | 5900 |

Massa de Dados

cat1=4

cat2=9

**17. Calcule a quantidade de litros gastos em uma viagem, sabendo que o automóvel faz 12 km/l. Receber o tempo de percurso e a velocidade média.**

Pseudo-Código

Algoritmo velocidade

Declarar.

h,vm,ql,t: num;

Início.

Ler(t);

Ler(vm);

ql=(vm\*t)/12;

Mostra (ql);

Fim

Teste de Mesa

|  |  |
| --- | --- |
| t | 2 |
| vm | 24 |
| ql | 4 |

Massa de Dados

t=42

vm=24

**EXERCÍCIOS DE ALGORITMO  
QUE SERÃO USADOS NO 2º SEM/2012 PARA LP**

**Lote 1**

1. Receba 2 valores inteiros. Calcule e mostre o resultado da diferença do maior pelo menos valor.

diferenca

X,Y: num

X,Y

Sim

Não

Se (X>=Y)

X=Y-X

X=X-Y

X

FIM

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

nnnmmmmmmmmmmmmmmmmmmm

Algoritmo diferenca

Declarar

X,Y: int;

Inicio

Ler(X);

Ler(Y);

Se(X>=Y) então

X=X-Y;

Senão

X=Y-X;

Fim-se;

Mostra (X);

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 1 | 2 | 3 | 3 |
| Y | 1 | 4 | 2 | 13 |

Massa de Dados Saída

1 1 0

2 4 2

3 2 1

3 3 0

1. Receba 2 valores reais. Calcule e mostre o maior deles.

Algoritmo maior

Declarar

X,Y: int;

Inicio

Ler(X);

Ler(Y);

Se(X>=Y) então

Mostra(X);

Senão

Mostra(Y);

Fim-se;

Fim;

1. Receba 3 coeficientes A, B, e C de uma equação do 2º grau da fórmula AX²+BX+C=0. Verifique e mostre a existência de raízes reais e se caso exista, calcule e mostre.

Algoritmo Bhaskara

Declarar

a,b,c,d,x1,x2:num;

Inicio

Ler(a);

Ler(b);

Ler(c);

Se(a==0) então

Mostra(“não é uma equação do segundo grau”);

Senão

d=(b\*b)-4\*A\*C;

se(d<0) então

mostra(”não há raíz real”);

senão

se(d==0)

x1=-b/2.a;

x2=x1;

mostra(“há apenas uma raiz real”)

senão

x1=-b+raiz(d)/2.a;

x1=-b-raiz(d)/2.a;

fim-se;

mostra(x1);

mostra(x2);

fim-se;

fim-se;

Fim;

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | 1 | 1 | 4 | 1 |
| B | 3 | 1 | 4 | 5 |
| C | 4 | 1 | 1 | 6 |
| D |  | -3 | 0 | 1 |
| x1 |  |  | -8 | -2 |
| x2 |  |  | -8 | -3 |

Massa de Dados Saída

0 3 4 “não é uma equação do segundo grau”

1 1 1 “não há raiz real”

4 4 1 “há apenas uma raiz real” -8 -8

1 5 6 -2, -3

1. Receba 4 notas bimestrais de um aluno. Calcule e mostre a média aritmética. Mostre a mensagem de acordo com a média:
   1. Se a média for >= 6,0 exibir “APROVADO”;
   2. Se a média for >= 3,0 ou < 6,0 exibir “EXAME”;
   3. Se a média for < 3,0 exibir “RETIDO”.

Algoritmo media

Declarar

x1,x2,x3,x4,media: num;

Inicio

Ler(x1);

Ler(x2);

Ler(x3);

Ler(x4);

Media=(x1+x2\_x3+x4)/4;

Se(media<3) então

Mostra (“RETIDO”);

Senão

Se(media>=6) então

Mostra(“APROVADO”)

Senão

Mostra(“EXAME”)

Fim-se;

Fim-se;

Fim;

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | 0 | 0 | 6 | 4 |
| x2 | 3 | 3 | 6 | 8 |
| x3 | 4 | 4 | 6 | 6 |
| x4 | 0 | 7 | 6 | 10 |
| Media | 1,75 | 3,5 | 6 | 7 |

Massa de Dados Saída

0 3 4 0 “RETIDO”

0 3 4 7 “EXAME”

6 6 6 6 “APROVADO”

4 8 6 10 “APROVADO”

1. Receba 2 valores inteiros e diferentes. Mostre seus valores em ordem crescente.

Algoritmo crescente\_2

Declarar

x,y: int;

Inicio

Ler(x);

Ler(y);

Se(x<y) então

Mostra (x);

Mostra (y);

Senão

Mostra (y);

Mostra (x);

Fim-se;

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 0 | 0 | 6 | 4 |
| Y | 3 | 3 | 6 | 8 |

Massa de Dados Saída

0 3 0 3

3 0 0 3

1. Receba 3 valores obrigatoriamente em ordem crescente e um 4º valor não necessariamente em ordem. Mostre os 4 números em ordem crescente.

Algoritmo crescente\_4

Declarar

x1,x2,x3,y: int;

Inicio

Ler(x1);

Ler(x2);

Ler(x3);

Ler(y)

Se(x1<y) então

Mostra (y);

Mostra (x1);

Mostra (x2);

Mostra (x3);

Senão

Mostra (x1);

Se(x2>y) então

Mostra (y);

Mostra (x2);

Mostra (x3);

Senão

Se (x3>y) então

Mostra(y);

Mostra(x3);

Senão

Mostra(x3);

Mostra(y);

Fim-se;

Fim-se;

Fim-se;

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | 0 | 2 | 4 | -1 |
| x2 | 0 | 2 | 4 | 1 |
| x3 | 0 | 2 | 4 | 3 |
| Y | 0 | 2 | 4 | 10 |

Massa de Dados Saída

0 2 4 -1 -1 0 2 4

0 2 4 1 0 1 2 4

0 2 4 3 0 2 3 4

0 2 4 10 0 2 4 10

1. Receba um valor inteiro. Verifique e mostre se é divisível por 2 e 3.

Algoritmo div\_2

Declarar

x,,r1,r2 :int;

Inicio

Ler(x);

r1=x%2;

r2=x%3;

Se(r1==0&&r2==0)

Mostra(x ”“édivisível por 2 e 3”);

Senão

Se(r1==0)

Mostra(x“é divisível por 2”)

Senão

Se(r2==0)

Mostra(x “é divisível por 3”);

Senão

Mostra(x “não é divisível por 2 e nem por 3”);

Fim-se;

Fim-se;

Fim-se;

Fim

1. Receba a hora de início e de final de um jogo (HH,MM) sabendo que o tempo máximo é de 24 horas e pode começar num dia e terminar noutro.

Algoritmo jogo

Declarar

hi, hF,mi, mF, tempoh, tempom :int

Inicio

Escrever(“Digite o horário inicial do jogo”)

Ler (hi);

Ler (mi);

Escrever (“Digite o horário final do jogo”)

Ler (hF);

Ler (mF);

Se (hF>=hi)

tempoh= hF-hi;

senão

tempoh= hF+24-hi;

fim-se;

Se(mF>=mi)

tempom= mF-mi;

Senão

tempom= mF+59-mi;

fim-se;

Se (hi<=0 || hF<=0 || hi>24 || hF>24 || mi<=0 ||mF<=0 || mi>59 || mF>59)

Mostra (”ERRO valor inválido”);

Senão

Mostra (“o Jogo durou” tempom “horas e ” tempom “minutos”);

Fim-se;

Fim

Teste de Mesa

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Hi | 0 | 22 | 4 | 20 | 30 |
| Mi | 0 | 0 | 4 | 50 | 10 |
| hF | 2 | 2 | 8 | 2 | 10 |
| mF | 50 | 50 | 0 | 10 | 10 |
| Tempoh | 2 | 4 | 4 | 6 | 20 |
| Tempom | 50 | 50 | 55 | 19 | 0 |

Massa de Dados Saída

0 0 2 50 “o Jogo durou 2 horas e 50 minutos”

22 0 2 50 “o Jogo durou 4 horas e 50 minutos”

4 4 8 0 “o Jogo durou 4 horas e 55 minutos”

20 50 2 10 “o Jogo durou 6 horas e 19 minutos”

30 10 10 10 “ERRO valor inválido”

1. Receba 2 números inteiros. Verifique e mostre se o maior número é múltiplo do menor.

Algoritmo mult

Declarar

x,y,r :int

Inicio

Ler(x);

Ler(y);

Se(x<y)

r=y%x;

se(r==0)

mostra(“é divisível”);

Senão

Mostra (“não é divisivel”);

Fim-se;

Senão

r=x%y;

se(r==0)

mostra(“é divisível”);

Senão

Mostra (“não é divisivel”);

Fim-se;

Fim-se;

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | 8 | 2 | 5 | 2 |
| Y | 4 | 4 | 2 | 5 |
| R | 0 | 0 | 1 | 1 |

Massa de Dados Saída

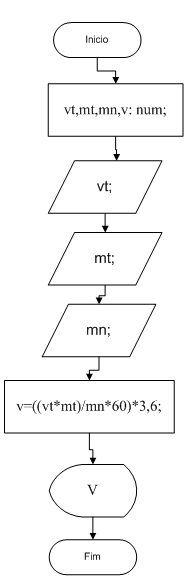
8 4 “é divisível”

2 4 “é divisível”

5 2 “não é divisivel”

2 5 “não é divisivel”

1. Receba o número de voltas, a extensão do circuito (em metros) e o tempo de duração (minutos). Calcule e mostre a velocidade média em km/h.



Algoritmo voltas

Declarar

vt,mt,mn,v: num;

Inicio

Ler(vt);

Ler(mt);

Ler(mn);

v=((vt\*mt)/mn\*60)\*3,6;

mostra(v);

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vt | 10 | 2 | 77 | 90 |
| Mt | 5 | 9000 | 208 | 140 |
| Mn | 100 | 50000 | 27655 | 21600 |
| V | 108 | 77,76 | 125,10 | 160,16 |

Massa de Dados

10 2 77 90

5 90.00 208 140

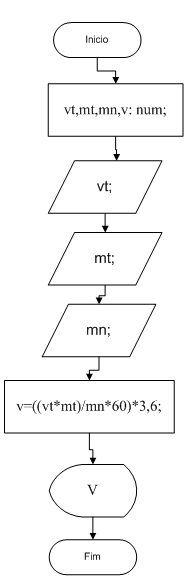
100 50.000 27.655 21.600

1. Receba o preço atual e a média mensal de um produto. Calcule e mostre o novo preço sabendo que:  
   Venda Mensal Preço Atual Preço Novo  
   < 500 < 30 + 10%  
   >= 500 e < 1000 >= 30 e < 80 +15%  
   >= 1000 >= 80 - 5%  
   Obs.: para outras condições, preço novo será igual ao preço atual.
2. Receba o tipo de investimento (1 = poupança e 2 = renda fixa) e o valor do investimento. Calcule e mostre o valor corrigido em 30 dias sabendo que a poupança = 3% e a renda fixa = 5%. Demais tipos não serão considerados.

**EXERCÍCIOS DE ALGORITMO  
QUE SERÃO USADOS NO1º SEM/2013**

**Lote 1.3**

1. Receba o número de voltas, a extensão do circuito (em metros) e o tempo de duração (minutos). Calcule e mostre a velocidade média em km/h.



Algoritmo voltas

Declarar

vt,mt,mn,v: num;

Inicio

Ler(vt);

Ler(mt);

Ler(mn);

v=((vt\*mt)/mn\*60)\*3,6;

mostra(v);

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vt | 10 | 2 | 77 | 90 |
| Mt | 5 | 9000 | 208 | 140 |
| Mn | 100 | 50000 | 27655 | 21600 |
| V | 108 | 77,76 | 125,10 | 160,16 |

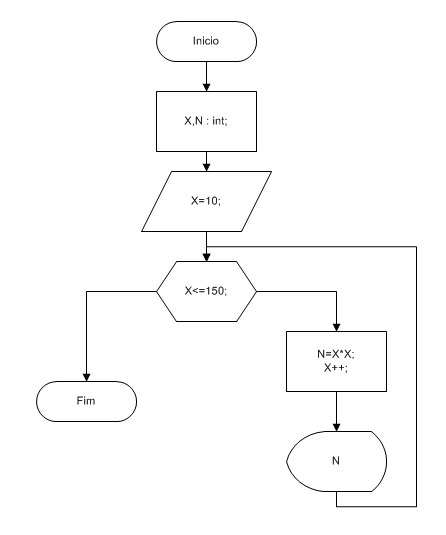
Massa de Dados

10 2 77 90

5 90.00 208 140

100 50.000 27.655 21.600

1. Receba o preço atual e a média mensal de um produto. Calcule e mostre o novo preço sabendo que:  
   Venda Mensal Preço Atual Preço Novo  
   < 500 < 30 + 10%  
   >= 500 e < 1000 >= 30 e < 80 +15%  
   >= 1000 >= 80 - 5%  
   Obs.: para outras condições, preço novo será igual ao preço atual.
2. Receba o tipo de investimento (1 = poupança e 2 = renda fixa) e o valor do investimento. Calcule e mostre o valor corrigido em 30 dias sabendo que a poupança = 3% e a renda fixa = 5%. Demais tipos não serão considerados.
3. Receba a data de nascimento e atual em ano, mês e dia. Calcule e mostre a idade em anos, meses e dias, considerando os anos bissextos.
4. Calcule e mostre o quadrado dos números entre 10 e 150.



Algoritmo Quadrados

Declarar

X=10,N: int;

Inicio

Enquanto(X<=150) faça

N=X\*X;

X++;

Mostra(N);

Fim-enquanto;

Fim

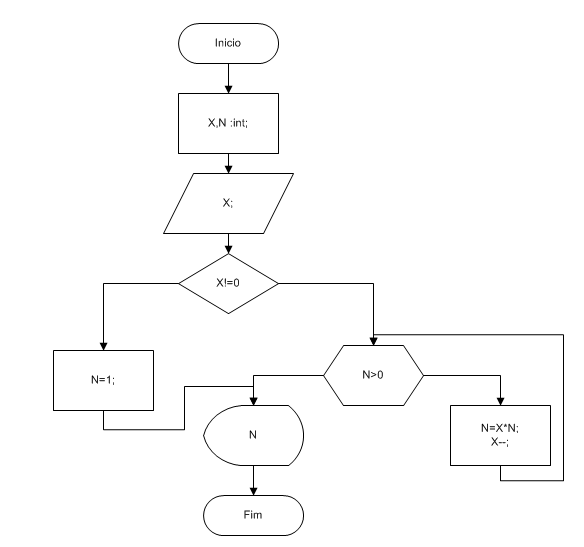
Teste de Mesa

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 10 | 11 | 12 | 13 | ... | 150 |
| N | 100 | 121 | 144 | 169 | ... | 22500 |

Massa de Dados

10

1. Receba um número inteiro. Calcule e mostre o seu fatorial.



Não

Sim

Não

Sim

Algoritmo Fatorial

Declarar

X,N: int;

Inicio

Ler(X);

Se(x!=0) então

Enquanto(N>0) faça

N=X\*N;

X--;

Fim-enquanto;

Senão

N=1;

Fim-se;

Mostra(N);

Fim

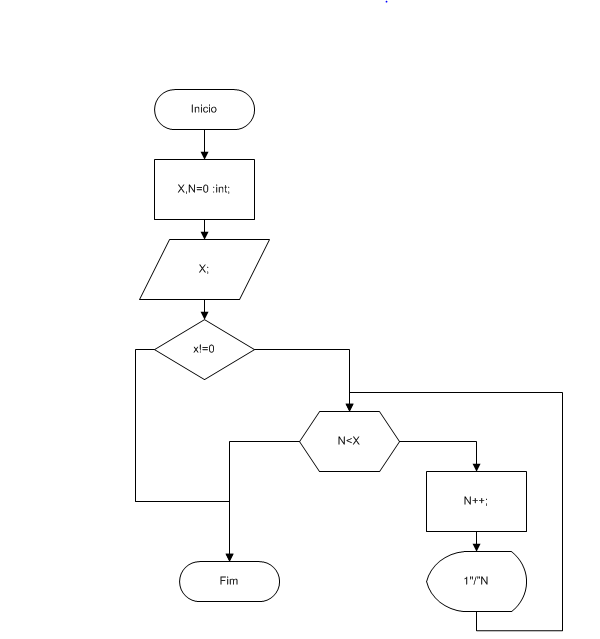
Teste de Mesa

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | 1 | 2 | 3 | 4 | 5 |
| N | 1 | 1 | 4 | 6 | 24 | 120 |

Massa de Dados

12 3 4 5

1. Receba um número. Calcule e mostre a série 1 + 1/2 + 1/3 + ... + 1/N.



Algoritmo Fatorial

Declarar

X,N: int;

Inicio

N=0;

Ler(X);

Mostrar(“1”);

Se (x!=0) então

Enquanto (N<=0) faça

N++;

Mostra(1”/”N;

Fim-enquanto;

Fim-se;

Fim

Teste de Mesa

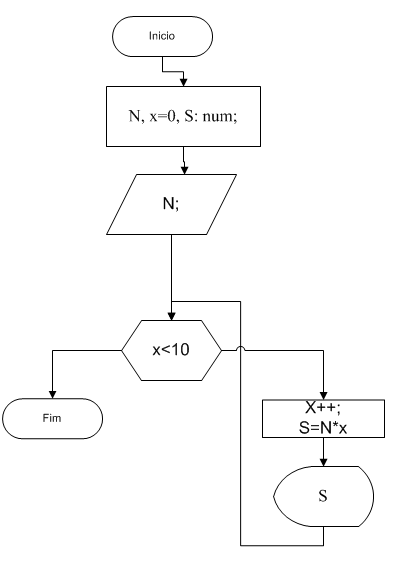
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | 5 | 5 | 5 | 5 | 5 |
| N |  | 1 | 2 | 3 | 4 | 5 |

Massa de Dados Saída

0

5 1/1 1/2 1/3 1/4 1/5

1. Receba um número. Calcule emostre os resultados da tabuada desse número.



Algoritmo Tabuada

Declarar

x=0,N, S :num;

Inicio

Ler(N);

Mostra(N);

Enquanto (X<0) faça

X++;

S=N\*x;

Mostra(S);

Fim-enquanto;

Fim

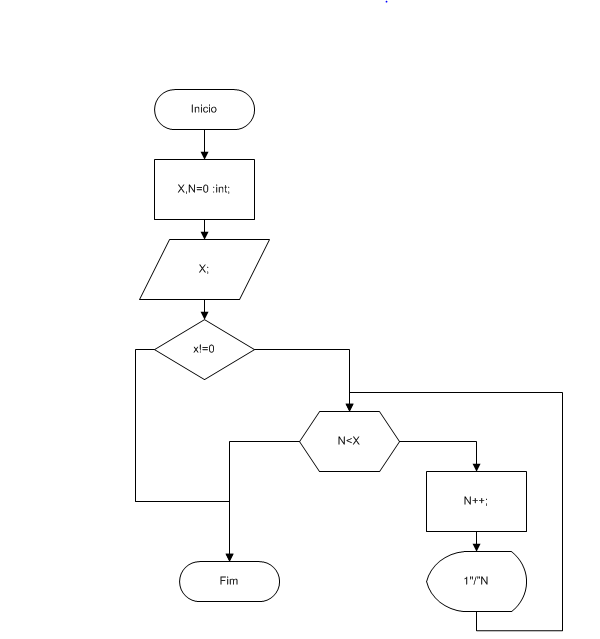
Teste de Mesa

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| N | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| S |  | 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 | 30 |

Massa de Dados

3

1. Receba 2 números inteiros, verifique qual o maior entre eles. Calcule e mostre o resultado da somatória dos números ímpares entre esses valores.
2. Receba um número N. Calcule e mostre a série 1 + 1/1! + 1/2! + ... + 1/N!



Algoritmo Fatorial

Declarar

X,N: int;

Inicio

N=0;

Ler(X);

Mostrar(“1”);

Se (x!=0) então

Enquanto (N<=0) faça

N++;

Mostra(1”/”N;

Fim-enquanto;

Fim-se;

Fim

Teste de Mesa

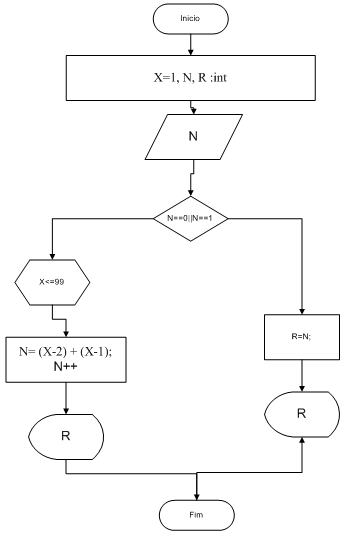
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | 5 | 5 | 5 | 5 | 5 |
| N |  | 1 | 2 | 3 | 4 | 5 |

Massa de Dados Saída

0

5 1/1 1/2 1/3 1/4 1/5

1. Receba um número inteiro. Calcule e mostre a série de Fibonacci até o seu N’nésimo termo.



Algoritmo Fatorial

Declarar

X,N : int;

Inicio

Ler(X);

Se (X==0 ||X==1) então

Mostra(X);

Se(X==0)

X=1;

Fim-se;

X=2;

Fim-se;

Enquanto (X<=99) faça

N= (X-2) + (X-1)// fibonacci[i]<- fibonacci[i-2]+ fibonacci[i-1]

X++;

Mostra(N);

Fim-enquanto;

Fim

Teste de Mesa

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N |  |  |  |  |  |  |
| X |  |  |  |  |  |  |
| R |  |  |  |  |  |  |

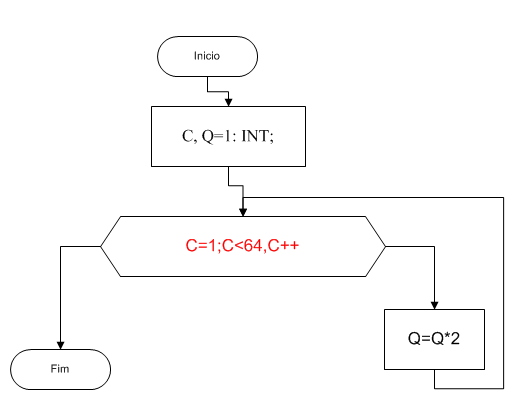
Massa de Dados

0

1

2

1. Receba 100 números inteiros reais. Verifique e mostre o maior e o menos valor. Obs.: somente valores positivos.
2. Calcule a quantidade de grãos contidos em um tabuleiro de xadrez onde:  
   Casa: 1 2 3 4 ... 64  
   Qte: 1 2 4 8 ... N



Algoritmo Fatorial

Declarar

C,Q=1 : int;

Inicio

Para(C==; C<64;c++) faça

Q=Q\*2;

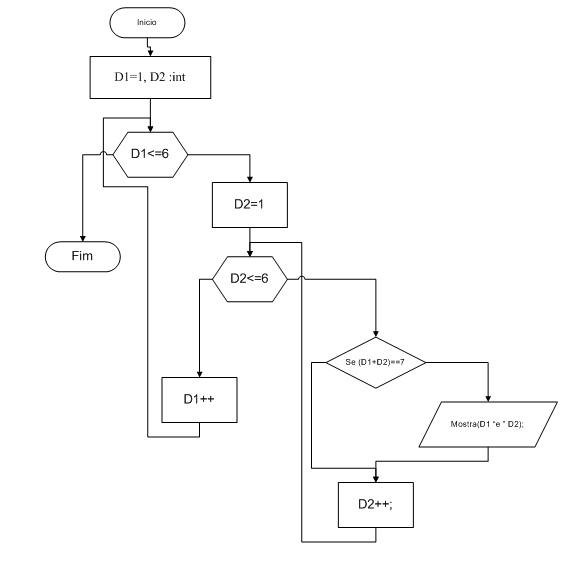
Fim-para;

Fim

Teste de Mesa

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| C | 1 | 2 | 3 | 4 | .... | 64 |
| Q | 1 | 2 | 4 | 8 | ... | 9,22337E+18 |

1. Receba 2 números inteiros. Verifique e mostre todos os números primos existentes entre eles.
2. Mostre todas as possibilidades de 2 dados de forma que a soma tenha como resultado 7.



Algoritmo Serie\_fracao

Declarar

D1=1, D2: int;

Inicio

Enquanto (D1<=6) faça

D2=1;

Enquanto(D1<=6)

Se((D1+D2)==7)

Mostra(D1” e “ D2);

fim-se;

D2++;

fim-Enquanto;

D1++;

Fim-enquanto;

Fim

Teste de Mesa

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| D1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | ... | 6 |
| D2 | 1 | 2 | 3 | 4 | 5 | 6 | 1 | 2 | 3 | 4 | 5 | 6 | ... | 6 |

Saída

1 e 6;

2 e 5;

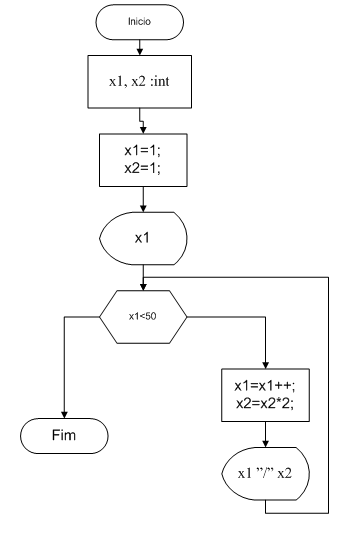
3 e 4;

4 e 3;

5 e 2;

1 e 6;

1. Calcule e mostre a série 1 + 2/3 + 3/5 + ... + 50/99



Algoritmo Serie\_fracao

Declarar

x1,x2: int;

Inicio

x1=1;

x2=1;

Mostra(x1);

Enquanto(x1<50)

x1=x1++;

x2=x2+2;

mostra(x1 ”/” x2);

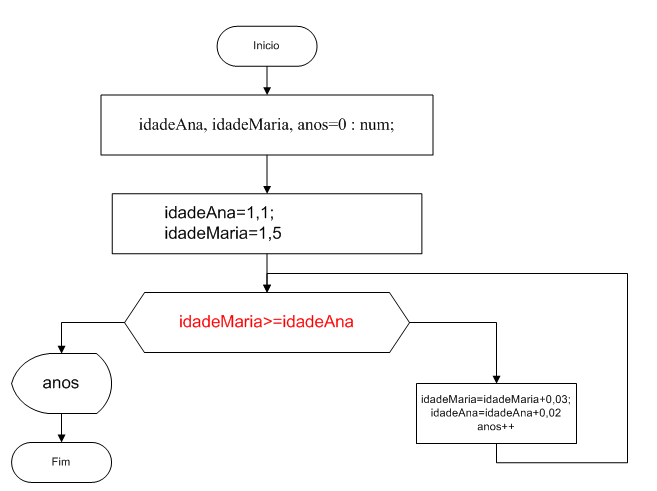
Fim-enquanto;

Fim

Teste de Mesa

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| x1 | 1 | 2 | 3 | 4 | 5 | .... | 50 |
| x2 | 1 | 3 | 5 | 7 | 9 | ... | 99 |

1. Calcule e mostre quantos anos serão necessários para que Ana seja maior que Maria sabendo que Ana tem 1,10 m e cresce 3 cm ao ano e Maria tem 1,5 m e cresce 2 cm ao ano.



Algoritmo Idade

Declarar

idadeAna,idadeMaria, anos=0 : num;

Inicio

idadeAna=1,10;

idadeMaria=1,5;

Enquanto(idadeMaria>=idadeAna)

idadeMaria=idadeMaria+0,03;

idadeAna=idadeAna+0,02;

anos=anos++;

Fim-enquanto;

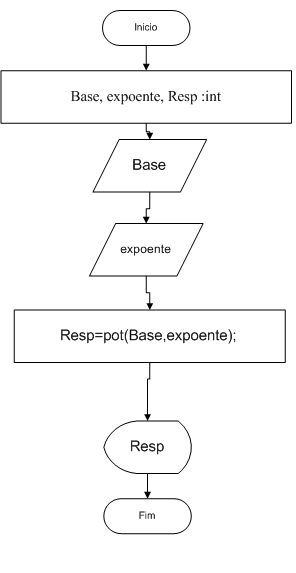
Mostra(anos);

Fim

Teste de Mesa

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| idadeAna | 1,10 | 1,13 | 1,16 | 1,19 | 1,22 | ... | 2,33 |
| Q | 1,5 | 1,52 | 1,54 | 1,56 | 1,58 | ... | 2,32 |
| anos | 1 | 2 | 3 | 4 | 5 | ... | 42 |

1. Receba o número da base e do expoente. Calcule e mostre o valor da potência



Algoritmo Potencia\_exp

Declarar

Base, expoente, Resp :int;

Inicio

Ler(Base);

Ler(expoente);

Resp= pot(Base, expoente);

Mostrar(resp);

Fim

Teste de Mesa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vt | 1 | 2 | 3 | 4 |
| Mt | 2 | 5 | 9 | 2 |
| Resp | 2 | 32 | 19863 | 16 |

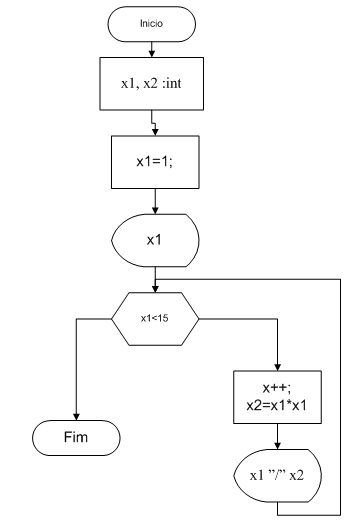
Massa de dados

12 3 4

25 9 2

.

1. Calcule e mostre a série 1 – 2/4 + 3/9 – 4/16 + 5/25 + ... + 15/225



Algoritmo Serie\_fracao

Declarar

x1,x2: int;

Inicio

x1=1;

Mostra(x1);

Enquanto(x1<15)

x1++;

x2=x1\*x1;

mostra(x1 ”/” x2);

Fim-enquanto;

Fim

Teste de Mesa

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| x1 | 1 | 2 | 3 | 4 | 5 | .... | 15 |
| x2 |  | 4 | 9 | 16 | 25 | ... | 225 |