



Lista de Exercícios 03a – Algoritmos – Repetição

- 5) Escreva um algoritmo em PORTUGOL que imprima todos os números inteiros de 200 a 100 (em ordem decrescente).

```
algoritmo L3P005;  
var  
  inteiro: C;  
início  
  para C de 200 até 100 passo -1 faça  
    imprima(C);  
  fim-para  
fim
```

```
program l3p005;  
var  
  C: integer;  
begin  
  for C := 200 downto 100 do  
    writeln(C);  
  end.
```

```
for C = 200 : -1 : 100  
  disp(C);  
end
```

- 10) Escreva um algoritmo em PORTUGOL que imprima todos os números ímpares do intervalo fechado de 1 a 100.

```
algoritmo L3P010;  
var  
  inteiro: C;  
início  
  para C de 1 até 100 passo 2 faça  
    imprima(C);  
  fim-para  
fim
```

```
program l3p010;  
var  
  C: integer;  
begin  
  for C := 1 to 100 do  
    if ( C mod 2 = 1) then  
      writeln(C);  
    end if;  
  end for;  
end.
```

```
for C = 1 : 100  
  if ( mod( C , 2 ) == 1 )  
    disp(C);  
  end if;  
end  
end
```



- 15) Escreva um algoritmo em PORTUGOL que receba oito números do usuário e imprima o logaritmo de cada um deles na base 10.

```
algoritmo L3P015;
var
  inteiro: C;
  real: VALOR, LOGX;
início
  para C de 1 até 8 faça
    leia (VALOR);
    se ( VALOR > 0 ) então
      LOGX <- log(VALOR) / log(10);
      imprima(LOGX);
    senão
      imprima("Não é possível calcular o log na base dez! Número negativo!");
    fim-se
  fim-para
fim

program l3p015;
var
  C: integer;
  VALOR, LOGX: real;
begin
  for C := 1 to 8 do
    begin
      write('Digite um valor: ');
      readLn(VALOR);
      if ( VALOR > 0 ) then begin
        LOGX := Ln(VALOR) / Ln(10);
        writeln(LOGX:3:2); end
      else
        writeln('Nao eh possivel calcular o log na base dez! Numero negativo!');
      end;
    end;
  end.

for C = 1 : 8
  VALOR = input('Digite um valor: ');
  if ( VALOR > 0 )
    LOGX = log(VALOR) / log(10);
    disp(LOGX);
  else
    disp('Nao eh possivel calcular o log na base dez! Numero negativo!');
  end
end
end
```



20) Criar um algoritmo em PORTUGOL que imprima todos os números de 1 até 100, inclusive, e a média de todos eles.

```
algoritmo L3P020;
var
  inteiro: C,SOMA;
  real: MEDIA;
início
  SOMA ← 0;
  para C de 1 até 100 faça
    imprima(C);
    SOMA ← SOMA + C;
  fim-para
  MEDIA ← SOMA / 100;
  imprima("O valor da média é: ",MEDIA);
fim

program l3p020;
var
  C : integer;
  SOMA,MEDIA: real;
begin
  SOMA := 0;
  for C := 1 to 100 do
    begin
      writeln(C);
      SOMA := SOMA + C;
    end;
  MEDIA := SOMA / 100;
  writeln('O valor da media eh: ',MEDIA:3:2);
end.

SOMA = 0;
for C=1:100
  disp(C);
  SOMA = SOMA + C;
end
MEDIA = SOMA / 100;
fprintf(1,'O valor da media eh: %f\n',MEDIA);
```



25) Criar um algoritmo em PORTUGOL que leia dez números inteiros e imprima o maior e o segundo maior número da lista.

```
algoritmo L3P025;  
var  
  inteiro: C, VALOR, M1, M2;  
  { M1 é o primeiro maior, e M2 é o segundo maior }  
início  
  leia(VALOR);  
  M1 <- VALOR;  
  para C de 2 até 10 faça  
    leia(VALOR);  
    se ( VALOR > M1 ) então  
      M2 <- M1;  
      M1 <- VALOR;  
    senão  
      se ( VALOR > M2 ) ou ( C = 2 ) então  
        M2 <- VALOR;  
      fim-se  
    fim-se  
  fim-para  
  imprima("O primeiro maior é: ", M1);  
  imprima("O segundo maior é: ", M2);  
fim
```

```
program l3p025;  
var  
  C, VALOR, M1, M2: integer;  
  { M1 eh o primeiro maior, e M2 eh o segundo maior }  
begin  
  write('Digite um valor: ');  
  readLn(VALOR);  
  M1 := VALOR;  
  for C := 2 to 10 do  
    begin  
      write('Digite um valor: ');  
      readLn(VALOR);  
      if ( VALOR > M1 ) then begin  
        M2 := M1;  
        M1 := VALOR; end  
      else  
        if ( VALOR > M2 ) or ( C = 2 ) then  
          M2 := VALOR;  
        end;  
      writeln('O primeiro maior eh: ', M1);  
      writeln('O segundo maior eh: ', M2);  
    end.  
  
  % M1 eh o primeiro maior, e M2 eh o segundo maior }  
  VALOR = input('Digite um valor: ');  
  M1 = VALOR;  
  for C = 2 : 10  
    VALOR = input('Digite um valor: ');  
    if ( VALOR > M1 )  
      M2 = M1;  
      M1 = VALOR;  
    else  
      if ( VALOR > M2 ) | ( C == 2 )  
        M2 = VALOR;  
      end  
    end  
  end  
  end  
  fprintf(1, 'O primeiro maior eh: %d\n', M1);  
  fprintf(1, 'O segundo maior eh: %d\n', M2);
```



30) Escreva um algoritmo em PORTUGOL que leia 20 números e imprima a soma dos positivos e o total de números negativos.

```
algoritmo L3P030;
var
  inteiro: C, NNEG;
  real: VALOR, SOMAP;
início
  SOMAP <- 0;
  NNEG <- 0;
  para C de 1 até 20 faça
    leia (VALOR);
    se ( VALOR >= 0 )
      então SOMAP <- SOMAP + VALOR; { número positivo }
      senão NNEG <- NNEG + 1; { número negativo }
    fim-se
  fim-para
  imprima ("A soma dos número positivos digitados é: ",SOMAP);
  imprima ("A quantidade de números negativos digitados é: ",NNEG);
fim

program l3p030;
var
  C, NNEG: integer;
  VALOR, SOMAP: real;
begin
  SOMAP := 0;
  NNEG := 0;
  for C := 1 to 20 do
    begin
      write('Digite um numero: ');
      readLn(VALOR);
      if ( VALOR >= 0 ) then
        SOMAP := SOMAP + VALOR { numero positivo }
      else
        NNEG := NNEG + 1;      { numero negativo }
      end;
      writeLn('A soma dos numero positivos digitados eh: ',SOMAP:3:2);
      writeLn('A quantidade de numeros negativos digitados eh: ',NNEG);
    end.

  SOMAP = 0;
  NNEG = 0;
  for C = 1 : 20
    VALOR = input('Digite um numero: ');
    if ( VALOR >= 0 )
      SOMAP = SOMAP + VALOR; % numero positivo
    else
      NNEG = NNEG + 1;      % numero negativo
    end
  end
  fprintf(1,'A soma dos numero positivos digitados eh: %f\n',SOMAP);
  fprintf(1,'A quantidade de numeros negativos digitados eh: %d\n',NNEG);
```



- 35) Escreva um algoritmo em PORTUGOL que determine se dois valores inteiros e positivos A e B são **primos** entre si. (dois números inteiros são ditos primos entre si, caso não exista divisor comum aos dois números).

```
algoritmo L3P035;  
var  
  inteiro: A, B, C;  
  lógico: SIT;  
início  
  SIT <- verdadeiro;  
  leia(A,B);  
  C <- 2;  
  enquanto ( SIT ) e ( C <= A ) e ( C <= B ) faça  
    se (A mod C = 0) e (B mod C = 0) então  
      SIT <- falso;  
    fim-se  
    C <- C + 1;  
  fim-enquanto  
  se ( SIT ) então  
    imprima("A e B são primos entre si");  
  senão  
    imprima("A e B não são primos entre si");  
  fim-se  
fim
```

```
program l3p035;  
var  
  A, B, C: integer;  
  SIT: boolean;  
begin  
  SIT := true;  
  write('Entre com valor de A: ');  
  readLn(A);  
  write('Entre com valor de B: ');  
  readLn(B);  
  C := 2;  
  while ( SIT ) and ( C <= A ) and ( C <= B ) do  
    begin  
      if (A mod C = 0) and (B mod C = 0) then  
        SIT := false;  
      C := C + 1;  
    end;  
    if ( SIT ) then  
      writeLn('A e B sao primos entre si')  
    else  
      writeLn('A e B nao sao primos entre si');  
  end.
```

```
SIT = 1; % true  
A = input('Entre com valor de A: ');  
B = input('Entre com valor de B: ');  
C = 2;  
while ( SIT ) & ( C <= A ) & ( C <= B )  
  if ( mod(A,C) == 0 ) & (mod(B,C) == 0)  
    SIT = 0; % false  
  end  
  C = C + 1;  
end  
if ( SIT )  
  disp('A e B sao primos entre si');  
else  
  disp('A e B nao sao primos entre si');  
end
```



- 40) Escreva um algoritmo em PORTUGOL que calcule o m.d.c. (máximo divisor comum) entre A e B (número inteiros e positivos). Esses dois valores são passados pelo usuário através do teclado.

```
algoritmo L3P040;
var
  inteiro: A, B, C, MDC;
início
  leia(A,B);
  C <- 1;
  enquanto ( C <= A ) e ( C <= B ) faça
    se (A mod C = 0) e (B mod C = 0) então
      MDC <- C;
    fim-se
    C <- C + 1;
  fim-enquanto
  imprima("O m.d.c. de A e B (m.d.c.(A,B)) é: ",MDC);
fim

program l3p040;
var
  A, B, C, MDC: integer;
begin
  write('Digite um numero: ');
  readLn(A);
  write('Digite outro numero: ');
  readLn(B);
  C := 1;
  while ( C <= A ) and ( C <= B ) do
    begin
      if (A mod C = 0) and (B mod C = 0) then
        MDC := C;
      C := C + 1;
    end;
  writeLn('O m.d.c. de A e B (m.d.c.(A,B)) eh: ',MDC);
end.

A = input('Digite um numero: ');
B = input('Digite outro numero: ');
C = 1;
while ( C <= A ) & ( C <= B )
  if ( mod(A,C) == 0 ) & ( mod(B,C) == 0 )
    MDC = C;
  end
  C = C + 1;
end
fprintf(1,'O m.d.c. de A e B (m.d.c.(A,B)) eh: %d\n',MDC);
```



45) Seja a seguinte série:

1, 4, 4, 2, 5, 5, 3, 6, 6, 4, 7, 7, ...

Escreva um algoritmo em PORTUGOL que seja capaz de gerar os N termos dessa série. Esse número N deve ser lido do teclado.

```
algoritmo L3P045;
var
  inteiro: N, C;
inicio
  leia(N);
  para C de 1 até N faça
    se ( C mod 3 = 1 ) então
      imprima( ( C div 3 ) + 1 ); { 1º, 4º, 7º , ... }
    senão-se ( C mod 3 = 2 ) então
      imprima( ( C div 3 ) + 4 ); { 2º, 5º, 8º , ... }
    senão
      imprima( ( C div 3 ) + 3 ); { 3º, 6º, 9º , ... }
    fim-se
  fim-para
fim

program l3p045;
var
  N, C: integer;
begin
  write('Digite o numero de termos: ');
  read(N);
  for C := 1 to N do
    begin
      if ( C mod 3 = 1 ) then
        writeln( ( C div 3 ) + 1 )      { 1o, 4o, 7o , ... }
      else if ( C mod 3 = 2 ) then
        writeln( ( C div 3 ) + 4 ) { 2o, 5o, 8o , ... }
      else
        writeln( ( C div 3 ) + 3 ); { 3o, 6o, 9o , ... }
      end;
    end;
  end.

N = input('Digite o numero de termos: ');
for C = 1 : N
  if ( mod(C,3) == 1 )
    fprintf(1,'%d\n', floor( C / 3 ) + 1 ) % 1o, 4o, 7o , ...
  elseif ( mod(C,3) == 2 )
    fprintf(1,'%d\n', floor( C / 3 ) + 4 ); % 2o, 5o, 8o , ...
  else
    fprintf(1,'%d\n', floor( C / 3 ) + 3 ); % 3o, 6o, 9o , ...
  end
end
end
```




50) Faça um algoritmo que:

- leia um número real X do teclado;
- determine e imprima o seguinte somatório:

$$S = X - \frac{X}{1!} + \frac{X}{2!} - \frac{X}{3!} + \frac{X}{4!} + \dots$$

usando os 20 primeiros termos da série.

```
algoritmo L3P050;  
var  
  inteiro: I, J, FAT;  
  real: X, S;  
início  
  leia(X);  
  S <- 0;  
  para I de 1 até 20 faça  
    FAT <- 1;  
    para J de 2 até I-1 faça  
      FAT <- FAT * J;  
    fim-para  
    se ( I mod 2 = 0 ) senão  
      S <- S - X / FAT; { termo par }  
    senão  
      S <- S + X / FAT; { termo ímpar }  
    fim-se  
  fim-para  
  imprima("S = ",S);  
fim
```

```
program l3p050;  
var  
  I, J: integer;  
  X, FAT, S: real;  
begin  
  write('Digite X: ');  
  readLn(X);  
  S := 0;  
  for I := 1 to 20 do  
    begin  
      FAT := 1;  
      for J := 2 to I-1 do  
        FAT := FAT * J;  
      if ( I mod 2 = 0 ) then  
        S := S - X / FAT { termo par }  
      else  
        S := S + X / FAT; { termo ímpar }  
      end;  
    writeLn('S = ',S:5:4);  
  end.  
end.
```



Universidade Federal de Ouro Preto – UFOP
Instituto de Ciências Exatas e Biológicas – ICEB
Departamento de Computação – DECOM
Disciplina: Algoritmos e Estrutura de Dados I – CIC102
Professor: David Menotti (menottid@gmail.com)



```
X = input('Digite X: ');
S = 0;
for I = 1 : 20
    FAT = 1;
    for J = 2 : I - 1
        FAT = FAT * J;
    end
    if ( mod(I,2) == 0 )
        S = S - X / FAT; % termo par
    else
        S = S + X / FAT; % termo impar
    end
end
fprintf(1, 'S = %f\n', S);
```



55) Criar um algoritmo que leia o valor de N, imprima a sequência a seguir e o resultado.

$$N! / 0! - (N-1)! / 2! + (N-2)! / 4! - (N-3)! / 6! + \dots 0! / (2N)!$$

```

algoritmo L3P055;
var
    inteiro: I, J, N, FATN, FATD;
    real: SOMA, R;
início
    leia(N);
    SOMA <- 0;
    para I de 0 até N faça
        FATN <- 1;
        para J de 2 até N - I faça
            FATN <- FATN * J;
        fim-para
        FATD <- 1;
        para J de 2 até 2*I * 2*I faça
            FATD <- FATD * J;
        fim-para
        R <- FATN / (FATD ** 2);
        imprima(R);
        SOMA <- SOMA + R;
    fim-para
    imprima("Soma = ",SOMA);
fim

program l3p055;
var
    I, J, N: integer;
    SOMA, R, FATN, FATD: real;
begin
    write('Entre com o valor de N: ');
    read(N);
    SOMA := 0;
    for I := 0 to N do
        begin
            FATN := 1;
            for J := 2 to N - I do
                FATN := FATN * J;
            FATD := 1;
            for J := 2 to (2 * I) do
                FATD := FATD * J;
            R := FATN / (FATD * FATD);
            writeln(I+1,'o. Termo = ',R:17:16);
            SOMA := SOMA + R;
        end;
        writeln('Soma = ',SOMA:17:16);
    end.

N = input('Entre com o valor de N: ');
SOMA = 0;
for I = 0 : N
    FATN = 1;
    for J = 2 : N - I
        FATN = FATN * J;
    end
    FATD = 1;
    for J = 2 : (2 * I)
        FATD = FATD * J;
    end
    R = FATN / (FATD * FATD);
    fprintf(1,'%do. Termo = %f\n',I+1,R);
    SOMA = SOMA + R;
end
fprintf(1,'Soma = %f\n',SOMA);
    
```



- 60) Calcule o imposto de renda de um grupo de 10 contribuintes, considerando que os dados de cada contribuinte, número do CPF, número de dependentes e renda mensal são valores fornecidos pelo usuário. Para cada contribuinte será feito um desconto de 5% do salário mínimo por dependente.

Os valores da alíquota para cálculo do imposto são:

Renda Líquida	Alíquota
Até 2 salários mínimos	Isento
2 a 3 salários mínimos	5%
3 a 5 salários mínimos	10%
5 a 7 salários mínimos	15%
Acima de 7 salários mínimos	20%

Observe que deve ser fornecido o valor atual do salário mínimo para que o algoritmo calcule os valores corretamente.

```
algoritmo L3P060;
var
  inteiro: C, CPF, NDEP;
  real: RENDA, SALMIN, IMP, NSALMIN;
início
  leia(SALMIN);
  para C de 1 até 10 faça
    leia(CPF);
    leia(NDEP);
    leia(RENDA);
    NSALMIN <- RENDA / SALMIN;
    se ( NSALMIN < 2 ) então
      IMP <- 0;
    senão-se ( NSALMIN < 3 ) então
      IMP <- RENDA*5/100;
    senão-se ( NSALMIN < 5 ) então
      IMP <- RENDA*10/100;
    senão-se ( NSALMIN < 7 ) então
      IMP <- RENDA*15/100;
    senão
      IMP <- RENDA*20/100;
  fim-se
  IMP <- IMP + NDEP * (SALMIN*5/100);
  se ( IMP > 0 ) então
    imprima("Imposto a ser pago: ", IMP);
  senão
    imprima("Imposto a ser restituído: ", -IMP);
  fim-se
fim-par
fim
```



```
program l3p060;
var
  C,CPF,NDEP: integer;
  RENDA,SALMIN,IMP,NSALMIN: real;
begin
  write('Informe o salario minimo: ');
  readLn(SALMIN);
  for C := 1 to 10 do
    begin
      write('CPF: ');
      readLn(CPF);
      write('N Dep: ');
      readLn(NDEP);
      write('Renda: ');
      readLn(RENDA);
      NSALMIN := RENDA / SALMIN;
      if ( NSALMIN < 2 ) then
        IMP := 0
      else if ( NSALMIN < 3 ) then
        IMP := RENDA*5/100
      else if ( NSALMIN < 5 ) then
        IMP := RENDA*10/100
      else if ( NSALMIN < 7 ) then
        IMP := RENDA*15/100
      else
        IMP := RENDA*20/100;
      IMP := IMP - NDEP * (SALMIN*5/100);
      if ( IMP > 0 ) then
        writeLn('Imposto a ser pago: ',IMP:3:2)
      else
        writeLn('Imposto a ser restituído: ',-IMP:3:2);
      end;
    end;
  end.

SALMIN = input('Informe o salario minimo: ');
for C = 1 : 10
  CPF = input('CPF: ');
  NDEP = input('N Dep: ');
  RENDA = input('Renda: ');
  NSALMIN = RENDA / SALMIN;
  if ( NSALMIN < 2 )
    IMP = 0;
  elseif ( NSALMIN < 3 )
    IMP = RENDA*5/100;
  elseif ( NSALMIN < 5 )
    IMP = RENDA*10/100;
  elseif ( NSALMIN < 7 )
    IMP = RENDA*15/100;
  else
    IMP = RENDA*20/100;
  end
  IMP = IMP - NDEP * (SALMIN*5/100);
  if ( IMP > 0 )
    fprintf(1,'Imposto a ser pago: %.2f\n',IMP);
  else
    fprintf(1,'Imposto a ser restituído: %.2f\n',-IMP);
  end
end
end
```