



Lista de Exercícios 03b – Algoritmos – Repetição

- 61) Escreva um algoritmo em PORTUGOL que receba números do usuário e imprima o triplo de cada número. O algoritmo deve encerrar quando o número -999 for digitado.

```
algoritmo L3P061;  
var  
  real : N;  
início  
  leia(N);  
  enquanto ( N ≠ -999 ) faça  
    imprima(N*3);  
    leia(N);  
  fim-enquanto  
fim
```

```
algoritmo L3P061B;  
var  
  real : N;  
início  
  repita:  
    leia(N);  
    se ( N ≠ -999 ) então  
      imprima(N*3);  
    fim-se  
  enquanto ( N ≠ -999 );  
fim
```

```
program l3p061;  
var  
  N: real;  
begin  
  write('Digite um numero: ');  
  readln(N);  
  while ( N <> -999 ) do  
    begin  
      writeln(N*3);  
      write('Digite um numero: ');  
      readln(N);  
    end;  
end.
```

```
program l3p061b;  
var  
  N: real;  
begin  
  repeat  
    write('Digite um numero: ');  
    readln(N);  
    if ( N <> -999 ) then  
      writeln(N*3);  
  until ( N = -999 );  
end.
```

```
% N: integer;  
N = input('Digite um numero: ');  
while ( N ~= -999 )  
  fprintf(1,N*3);  
  N = input('Digite um numero: ');  
end
```



- 62) Escreva um algoritmo em PORTUGOL que receba números do usuário enquanto eles forem positivos e ao fim o algoritmo deve imprimir quantos números foram digitados.

```
algoritmo L3P062;  
var  
  real : N;  
  inteiro : C;  
início  
  C ← 0 ;  
  leia(N);  
  enquanto ( N > 0 ) faça  
    C ← C + 1;  
    imprima(N*3);  
    leia(N);  
  fim-enquanto  
  imprima("Foram digitados ",C," números!");  
fim
```

```
algoritmo L3P062B;  
var  
  real : N;  
  inteiro : C;  
início  
  C ← 0;  
  repita:  
    leia(N);  
    se ( N > 0 ) então  
      C ← C + 1;  
    fim-se  
  enquanto ( N > 0 );  
  imprima("Foram digitados ",C," números!");  
fim
```

```
program l3p062;  
var  
  N : real;  
  C : integer;  
begin  
  C := 0 ;  
  write('Digite um numero: ');  
  readLn(N);  
  while ( N > 0 ) do  
  begin  
    C := C + 1;  
    write('Digite um numero: ');  
    readLn(N);  
  end;  
  writeLn('Foram digitados ',C,' numeros!');  
end.
```

```
program l3p062b;  
var  
  N : real;  
  C : integer;  
begin  
  C := 0;  
  repeat  
    write('Digite um numero: ');  
    readLn(N);  
    if ( N > 0 ) then  
      C := C + 1;  
  until ( N <= 0 );  
  writeLn('Foram digitados ',C,' numeros!');  
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)



```
% N, C: integer;  
C = 0 ;  
N = input('Digite um numero: ');  
while ( N > 0 )  
    C = C + 1;  
    N = input('Digite um numero: ');  
end  
fprintf(1,'Foram digitados %d numeros!',C);
```



- 63) Escreva um algoritmo em PORTUGOL que receba números do usuário enquanto eles forem positivos e ao fim o algoritmo deve imprimir a média dos números digitados.

```
algoritmo L3P063;  
var  
  inteiro: C;  
  real: N, SOMA, MEDIA;  
início  
  C ← 0;  
  SOMA ← 0;  
  leia(N);  
  enquanto ( N > 0 ) faça  
    C ← C + 1;  
    SOMA ← SOMA + N;  
    leia(N);  
  fim-enquanto  
  MEDIA ← SOMA / C;  
  imprima("A média dos valores digitados é: ",MEDIA);  
fim
```

```
algoritmo L3P063;  
var  
  inteiro : C;  
  real : N, SOMA, MEDIA;  
início  
  C ← 0;  
  SOMA ← 0;  
  repita:  
    leia(N);  
    se ( N > 0 ) então  
      C ← C + 1;  
      SOMA ← SOMA + N;  
    fim-se  
  enquanto ( N > 0 );  
  MEDIA ← SOMA / C;  
  imprima("A média dos valores digitados é: ",MEDIA);  
fim
```

```
program l3p063;  
var  
  C : integer;  
  N, SOMA, MEDIA : real;  
begin  
  C := 0;  
  SOMA := 0;  
  write('Digite um numero: ');  
  readln(N);  
  while ( N > 0 ) do  
  begin  
    C := C + 1;  
    SOMA := SOMA + N;  
    write('Digite um numero: ');  
    read(N);  
  end;  
  MEDIA := SOMA / C;  
  write('A media dos valores digitados eh: ',MEDIA:3:2);  
end.
```



```
program l3p063b;
var
  N, C, SOMA: integer;
  MEDIA: real;
begin
  C := 0;
  SOMA := 0;
  repeat
    write('Digite um numero: ');
    readLn(N);
    if ( N > 0 ) then begin
      C := C + 1;
      SOMA := SOMA + N; end;
  until ( N <= 0 );
  MEDIA := SOMA / C;
  write('A media dos valores digitados eh: ',MEDIA:3:2);
end.

% N,C,SOMA: integer;
% MEDIA: real;
C = 0;
SOMA = 0;
N = input('Digite um numero: ');
while ( N > 0 )
  C = C + 1;
  SOMA = SOMA + N;
  N = input('Digite um numero: ');
end
MEDIA = SOMA / C;
fprintf('A media dos valores digitados eh: %f\n',MEDIA);
```



- 64) Escreva um algoritmo em PORTUGOL que leia vários números e informe quantos números entre 100 e 200 foram digitados. Quando o valor 0 (zero) for lido, o algoritmo deverá cessar sua execução.

```
algoritmo L3P064;  
var  
  real : N;  
  inteiro : C;  
início  
  C ← 0;  
  leia(N);  
  enquanto ( N ≠ 0 ) faça  
    se ( N > 100 e N < 200 ) então  
      C ← C + 1;  
    fim-se  
  leia(N);  
fim-enquanto  
imprima("Foram digitados ",C," números entre 100 e 200!");  
fim
```

```
algoritmo L3P064B;  
var  
  real : N;  
  inteiro : C;  
início  
  C ← 0;  
  repita:  
    leia(N);  
    se ( N > 100 e N < 200 ) então  
      C ← C + 1;  
    fim-se  
  enquanto ( N ≠ 0 );  
imprima("Foram digitados ",C," números entre 100 e 200!");  
fim
```

```
program l3p064;  
var  
  C : integer;  
  N : real;  
begin  
  C := 0;  
  write('Digite um valor: ');  
  read(N);  
  while ( N <> 0 ) do  
    begin  
      if ( N > 100 ) and ( N < 200 ) then  
        C := C + 1;  
      write('Digite um valor: ');  
      read(N);  
    end;  
  write('Foram digitados ',C,' numeros entre 100 e 200!');  
end.
```

```
program l3p064b;  
var  
  C : integer;  
  N : real;  
begin  
  C := 0;  
  repeat  
    write('Digite um valor: ');  
    readLn(N);  
    if ( N > 100 ) and ( N < 200 ) then  
      C := C + 1;  
  until ( N = 0 );  
  writeLn('Foram digitados ',C,' numeros entre 100 e 200!');  
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)



```
%N,C: integer;  
C = 0;  
N = input('Digite um valor: ');  
while ( N ~= 0 )  
    if ( N > 100 ) & ( N < 200 )  
        C = C + 1;  
    end  
    N = input('Digite um valor: ');  
end  
fprintf(1,'Foram digitados %d numeros entre 100 e 200!\n',C);
```



- 65) Escreva um algoritmo em PORTUGOL que receba números e imprima o quadrado de cada número até entrar um número múltiplo de 6 que deverá ter seu quadrado também impresso.

```
algoritmo L3P065;  
var  
  inteiro: N;  
inicio  
  N ← 1; { N deve ser inicializado com qualquer valor que não seja múltiplo de 6 }  
  enquanto ( N mod 6 ) ≠ 0 ) faça  
    leia(N);  
    imprima(N**2);  
  fim-enquanto  
fim
```

```
algoritmo L3P065B;  
var  
  inteiro: N;  
inicio  
  repita:  
    leia(N);  
    imprima(N**2);  
  enquanto ( N mod 6 ) ≠ 0 );  
fim
```

```
program l3p065;  
var  
  N: integer;  
begin  
  N := 1; { N deve ser inicializado com qualquer valor que não seja múltiplo de 6 }  
  while ( N mod 6 <> 0 ) do  
    begin  
      write('Digite um numero: ');  
      readLn(N);  
      writeLn(N*N);  
    end;  
end.
```

```
program l3p065b;  
var  
  N: integer;  
begin  
  repeat  
    write('Digite um numero: ');  
    readLn(N);  
    writeLn(N*N);  
  until ( N mod 6 = 0 );  
end.
```

```
N = 1; % N deve ser inicializado com qualquer valor que não seja múltiplo de 6  
while ( mod(N,6) ~= 0 )  
  N = input('Digite um numero: ');  
  disp(N*N);  
end
```




- 66) Construa um algoritmo em PORTUGOL para calcular a média de valores PARES e ÍMPARES, que serão digitados pelo usuário. Ao final o algoritmo deve mostrar estas duas médias. O algoritmo deve mostrar também o maior número PAR digitado e o menor número ÍMPAR digitado. Para finalizar o usuário irá digitar um valor negativo.

```
algoritmo L3P066;  
var  
  inteiro: VALOR, SOMAPAR, SOMAIMP, MAIORPAR, MENORIMP, CPAR, CIMP;  
  real:     MEDIAPAR, MEDIAIMP;  
inicio  
  SOMAPAR ← 0;  
  SOMAIMP ← 0;  
  MAIORPAR ← 0;  
  MAIORPAR ← 0;  
  CPAR ← 0;  
  CIMP ← 0;  
  repita:  
    leia(VALOR);  
    se ( VALOR > 0 ) então  
      se ( VALOR mod 2 = 0 ) { é par } então  
        se ( ( VALOR > MAIORPAR ) ou ( CPAR = 0 ) ) então  
          MAIORPAR ← VALOR;  
        fim-se  
        SOMAPAR ← SOMAPAR + VALOR;  
        CPAR ← CPAR + 1;  
      senão  
        se ( ( VALOR < MENORIMP ) ou ( CIMP = 0 ) ) então  
          MENORIMP ← VALOR;  
        fim-se  
        SOMAIMP ← SOMAIMP + VALOR;  
        CIMP ← CIMP + 1;  
      fim-se  
    fim-se  
  enquanto ( VALOR > 0 );  
  MEDIAPAR ← SOMAPAR / CPAR;  
  MEDIAIMP ← SOMAIMP / CIMP;  
  imprima("A média dos valores pares digitados é: ",MEDIAPAR);  
  imprima("A média dos valores ímpares digitados é: ",MEDIAIMP);  
fim
```



```
algoritmo L3P066B;  
var  
inteiro: VALOR, SOMAPAR, SOMAIMP, MAIORPAR, MAIORPAR, CPAR, CIMP;  
real: MEDIAPAR, MEDIAIMP;  
início  
SOMAPAR ← 0;  
SOMAIMP ← 0;  
MAIORPAR ← 0;  
MAIORPAR ← 0;  
CPAR ← 0;  
CIMP ← 0;  
leia(VALOR);  
enquanto ( VALOR > 0 ) faça  
  se ( VALOR mod 2 = 0 ) { é par } então  
    se ( ( VALOR > MAIORPAR ) ou ( CPAR = 0 ) ) então  
      MAIORPAR ← VALOR;  
    fim-se  
    SOMAPAR ← SOMAPAR + VALOR;  
    CPAR ← CPAR + 1;  
  senão  
    se ( ( VALOR < MENORIMP ) ou ( CIMP = 0 ) ) então  
      MENORIMP ← VALOR;  
    fim-se  
    SOMAIMP ← SOMAIMP + VALOR;  
    CIMP ← CIMP + 1;  
  fim-se  
leia(VALOR);  
fim-enquanto  
MEDIAPAR ← SOMAPAR / CPAR;  
MEDIAIMP ← SOMAIMP / CIMP;  
imprima("A média dos valores pares digitados é: ",MEDIAPAR);  
imprima("A média dos valores ímpares digitados é: ",MEDIAIMP);  
fim
```

```
program l3p066;  
var  
  VALOR, SOMAPAR, SOMAIMP, MAIORPAR, MENORIMP, CPAR, CIMP: integer;  
  MEDIAPAR, MEDIAIMP: real;  
begin  
  SOMAPAR := 0;  
  SOMAIMP := 0;  
  MAIORPAR := 0;  
  MENORIMP := 0;  
  CPAR := 0;  
  CIMP := 0;  
  repeat  
    write('Digite um valor: ');  
    read(VALOR);  
    if ( VALOR >= 0 ) then  
      if ( VALOR mod 2 = 0 ) then begin{ eh par }  
        if ( VALOR > MAIORPAR ) or ( CPAR = 0 ) then  
          MAIORPAR := VALOR;  
        SOMAPAR := SOMAPAR + VALOR;  
        CPAR := CPAR + 1; end  
      else begin  
        if ( VALOR < MENORIMP ) or ( CIMP = 0 ) then  
          MENORIMP := VALOR;  
        SOMAIMP := SOMAIMP + VALOR;  
        CIMP := CIMP + 1; end;  
    until ( VALOR < 0 );  
    MEDIAPAR := SOMAPAR / CPAR;  
    MEDIAIMP := SOMAIMP / CIMP;  
    writeln('A media dos valores pares digitados eh: ',MEDIAPAR:5:4);  
    writeln('A media dos valores impares digitados eh: ',MEDIAIMP:5:4);  
    writeln('O maior numero par digitado eh: ',MAIORPAR);  
    writeln('O menor numero impar digitado eh: ',MENORIMP);  
end.
```



```
program l3p066b;
var
  VALOR, SOMAPAR, SOMAIMP, MAIORPAR, MENORIMP, CPAR, CIMP: integer;
  MEDIAPAR, MEDIAIMP: real;
begin
  SOMAPAR := 0;
  SOMAIMP := 0;
  MAIORPAR := 0;
  MENORIMP := 0;
  CPAR := 0;
  CIMP := 0;
  write('Digite um valor: ');
  readLn(VALOR);
  while ( VALOR >= 0 ) do
  begin
    if ( VALOR mod 2 = 0 ) then begin { é par }
      if ( ( MAIORPAR < VALOR ) or ( CPAR = 0 ) ) then
        MAIORPAR := VALOR;
      SOMAPAR := SOMAPAR + VALOR;
      CPAR := CPAR + 1; end
    else begin
      if ( ( MENORIMP > VALOR ) or ( CIMP = 0 ) ) then
        MENORIMP := VALOR;
      SOMAIMP := SOMAIMP + VALOR;
      CIMP := CIMP + 1; end;
      write('Digite um valor: ');
      readLn(VALOR);
    end;
    MEDIAPAR := SOMAPAR / CPAR;
    MEDIAIMP := SOMAIMP / CIMP;
    writeLn('A media dos valores pares digitados eh: ',MEDIAPAR:5:4);
    writeLn('A media dos valores impares digitados eh: ',MEDIAIMP:5:4);
    writeLn('O maior numero par digitado eh: ',MAIORPAR);
    writeLn('O menor numero impar digitado eh: ',MENORIMP);
  end.

% VALOR, SOMAPAR, SOMAIMP, MAIORPAR, MENORIMP, CPAR, CIMP: integer;
% MEDIAPAR, MEDIAIMP: real;
SOMAPAR = 0;
SOMAIMP = 0;
MAIORPAR = 0;
MENORIMP = 0;
CPAR = 0;
CIMP = 0;
VALOR = input('Digite um valor: ');
while ( VALOR >= 0 )
  if ( VALOR >= 0 )
    if ( mod(VALOR,2) == 0 ) % eh par
      if ( VALOR > MAIORPAR ) | ( CPAR == 0 )
        MAIORPAR = VALOR;
      end
      SOMAPAR = SOMAPAR + VALOR;
      CPAR = CPAR + 1;
    else
      if ( VALOR < MENORIMP ) | ( CIMP == 0 )
        MENORIMP = VALOR;
      end
      SOMAIMP = SOMAIMP + VALOR;
      CIMP = CIMP + 1;
    end
  end
  VALOR = input('Digite um valor: ');
end
MEDIAPAR = SOMAPAR / CPAR;
MEDIAIMP = SOMAIMP / CIMP;
fprintf(1,'A media dos valores pares digitados eh: %f\n',MEDIAPAR);
fprintf(1,'A media dos valores impares digitados eh: %f\n',MEDIAIMP);
fprintf(1,'O maior numero par digitado eh: %d\n',MAIORPAR);
fprintf(1,'O menor numero impar digitado eh: %d\n',MENORIMP);
```



- 67) Construa um algoritmo em PORTUGOL para fazer a soma de vários valores inteiros e positivos, fornecidos pelo usuário através do teclado. O dado que finaliza a seqüência de entrada é o número -1, e este não deve ser considerado.

```
algoritmo L3P067;  
var  
  inteiro: VALOR, SOMA;  
inicio  
  SOMA ← 0;  
  repita:  
    leia(VALOR);  
    se ( VALOR > 0 ) então  
      SOMA ← SOMA + VALOR;  
    fim-se  
  enquanto ( VALOR ≠ -1 );  
  imprima("A soma dos valores digitados é: ",SOMA);  
fim
```

```
algoritmo L3P067B;  
var  
  inteiro: VALOR, SOMA;  
inicio  
  SOMA ← 0;  
  leia(VALOR);  
  enquanto( VALOR ≠ -1 ) faça  
    se ( VALOR > 0 ) então  
      SOMA ← SOMA + VALOR;  
    fim-se  
  leia(VALOR);  
  fim-enquanto  
  imprima("A soma dos valores digitados é: ",SOMA);  
fim
```

```
program l3p067;  
var  
  VALOR,SOMA: integer;  
begin  
  SOMA := 0;  
  repeat  
    write('Digite um valor: ');  
    readLn(VALOR);  
    if ( VALOR > 0 ) then  
      SOMA := SOMA + VALOR;  
  until ( VALOR = -1 );  
  writeLn('A soma dos valores digitados eh: ',SOMA);  
end.
```

```
program l3p067b;  
var  
  VALOR,SOMA: integer;  
begin  
  SOMA := 0;  
  write('Digite um valor: ');  
  readLn(VALOR);  
  while ( VALOR <> -1 ) do  
    begin  
      if ( VALOR > 0 ) then  
        SOMA := SOMA + VALOR;  
      write('Digite um valor: ');  
      readLn(VALOR);  
    end;  
  writeLn('A soma dos valores digitados eh: ',SOMA);  
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)



```
% VALOR,SOMA: integer;
SOMA = 0;
VALOR = input('Digite um valor: ');
while ( VALOR ~= -1 )
    if ( VALOR > 0 )
        SOMA = SOMA + VALOR;
    end
    VALOR = input('Digite um valor: ');
end
fprintf(1,'A soma dos valores digitados eh: %d\n',SOMA);
```



- 68) Construa um algoritmo em PORTUGOL para calcular a média de um conjunto de valores inteiros e positivos, fornecidos pelo usuário através do teclado. Novamente, o dado finalizar é o número -1, e este não deve ser considerado.

```
algoritmo L3P068;  
var  
  inteiro: VALOR, SOMA, CONT;  
  real: MEDIA;  
início  
  SOMA ← 0;  
  CONT ← 0;  
  leia(VALOR);  
  enquanto ( VALOR ≠ -1 ) faça  
    se ( VALOR > 0 ) então  
      SOMA ← SOMA + VALOR;  
      CONT ← CONT + 1;  
    fim-se  
  leia(VALOR);  
  fim-enquanto  
  MEDIA ← SOMA / CONT;  
  imprima("A média dos valores digitados é: ",MEDIA);  
fim
```

```
algoritmo L3P068B;  
var  
  inteiro: VALOR, SOMA, CONT;  
  real: MEDIA;  
início  
  SOMA ← 0;  
  CONT ← 0;  
  repita:  
    leia(VALOR);  
    se ( VALOR > 0 ) então  
      SOMA ← SOMA + VALOR;  
      CONT ← CONT + 1;  
    fim-se  
  enquanto ( VALOR ≠ -1 );  
  MEDIA ← SOMA / CONT;  
  imprima("A média dos valores digitados é: ",MEDIA);  
fim
```

```
program l3p068;  
var  
  VALOR,SOMA,CONT: integer;  
  MEDIA: real;  
begin  
  SOMA := 0;  
  CONT := 0;  
  write('Digite um valor: ');  
  readLn(VALOR);  
  while ( VALOR <> -1 ) do  
  begin  
    if ( VALOR > 0 ) then begin  
      SOMA := SOMA + VALOR;  
      CONT := CONT + 1; end;  
    write('Digite um valor: ');  
    readLn(VALOR);  
  end;  
  MEDIA := SOMA / CONT;  
  writeLn('A media dos valores digitados eh: ',MEDIA:3:4);  
end.
```



```
program l3p068b;
var
  VALOR,SOMA,CONT: integer;
  MEDIA: real;
begin
  SOMA := 0;
  CONT := 0;
  repeat
    write('Digite um valor: ');
    readLn(VALOR);
    if ( VALOR > 0 ) then begin
      SOMA := SOMA + VALOR;
      CONT := CONT + 1; end;
  until ( VALOR = -1 );
  MEDIA := SOMA / CONT;
  writeLn('A media dos valores digitados eh: ',MEDIA:3:4);
end.

% VALOR,SOMA,CONT: integer;
% MEDIA: real;
SOMA = 0;
CONT = 0;
VALOR = input('Digite um valor: ');
while ( VALOR ~= -1 )
  if ( VALOR > 0 )
    SOMA = SOMA + VALOR;
    CONT = CONT + 1;
  end
  VALOR = input('Digite um valor: ');
end
MEDIA = SOMA / CONT;
fprintf(1,'A media dos valores digitados eh: %f\n',MEDIA);
```



- 69) Construa um algoritmo em PORTUGOL para que: Dado um conjunto de valores inteiros e positivos, digitados pelo usuário através do teclado, determinar qual o menor valor do conjunto. O algoritmo deve imprimir o valor determinado. Quando o usuário decidir que finalizou a entrada de dados, ele deve digitar -1, e este dado não deve ser considerado.

```
algoritmo L3P069;  
var  
  inteiro: VALOR, MENOR;  
  lógico: PRI;  
inicio  
  PRI ← verdadeiro;  
  leia(VALOR);  
  enquanto ( VALOR <> -1) faça  
    se ( VALOR > 0 ) então  
      se ( VALOR < MENOR ou PRI) então  
        MENOR ← VALOR;  
        se ( PRI = verdadeiro ) então  
          PRI ← falso;  
        fim-se  
      fim-se  
    fim-enquanto  
  leia(VALOR);  
  se ( PRI = verdadeiro ) então  
    imprima("nenhum valor digitado válido!",MENOR);  
  senão  
    imprima("O menor valor digitado é: ",MENOR);  
  fim-se  
fim
```

```
program l3p069;  
var  
  VALOR, MENOR: integer;  
  PRI: boolean;  
begin  
  PRI := true;  
  MENOR := 0;  
  while ( VALOR <> -1 ) do  
    begin  
      write('Digite um valor: ');  
      readLn(VALOR);  
      if ( VALOR > 0 ) then begin  
        if ( VALOR < MENOR ) or ( PRI ) then  
          MENOR := VALOR;  
        if PRI then  
          PRI := false;  
        end;  
      end;  
    if ( PRI = true ) then  
      writeLn('nenhum valor digitado válido!');  
    else  
      writeLn('O menor valor digitado eh: ',MENOR);  
    end.  
end.
```




```
% VALOR, MENOR: integer;
VALOR = input('Digite um valor: ');
PRI = 1;
while ( VALOR ~= -1 )
    if ( VALOR > 0 )
        if ( VALOR < MENOR ) | (PRI == 1 )
            MENOR = VALOR;
            if (PRI == 1)
                PRI = 0;
            end
        end
    end
end
VALOR = input('Digite um valor: ');
end
if (PRI == 1)
    disp(1,'nenhum valor digitado válido!');
else
    fprintf(1,'O menor valor digitado eh: %d\n',MENOR);
end
```



- 70) Construa um algoritmo em PORTUGOL para encontrar o maior e o menor número de uma série de números positivos fornecidos pelo usuário através do teclado. Novamente, o dado finalizador é o número -1, e este não deve ser considerado.

```
algoritmo L3P070;  
var  
  real : VALOR, MENOR, MAIOR;  
  lógico : SIT;  
início  
  SIT ← verdadeiro;  
  leia(VALOR);  
  enquanto ( VALOR ≠ -1 ) faça  
    leia(VALOR);  
    se ( VALOR > 0 ) então  
      se ( SIT = verdadeiro )  
        SIT ← falso;  
        MENOR ← VALOR;  
        MAIOR ← VALOR;  
      fim-se  
      MENOR ← VALOR;  
      MAIOR ← VALOR;  
      se ( VALOR < MENOR ) então  
        MENOR ← VALOR;  
      senão-se ( VALOR > MAIOR ) então  
        MAIOR ← VALOR;  
      fim-se  
    fim-se  
  fim-enquanto  
  se ( SIT = verdadeiro )  
    imprima("nenhum valor digitado válido!");  
  senão  
    imprima("O menor valor digitado é: ",MENOR);  
    imprima("O maior valor digitado é: ",MAIOR);  
  fim-se  
fim
```

```
algoritmo L3P070B;  
var  
  real : VALOR, MENOR, MAIOR;  
  lógico : PRI;  
início  
  PRI ← verdadeiro;  
  MENOR ← 0;  
  MAIOR ← 0;  
  faça  
    leia(VALOR);  
    se ( VALOR > 0 ) então  
      se ( ( VALOR < MENOR ) ou ( PRI = verdadeiro ) ) então  
        MENOR ← VALOR;  
      fim-se  
      se ( ( VALOR > MAIOR ) ou ( PRI = verdadeiro ) ) então  
        MAIOR ← VALOR;  
      fim-se  
    PRI ← falso;  
  fim-se  
  enquanto ( VALOR ≠ -1 );  
  imprima("O menor valor digitado é: ",MENOR);  
  imprima("O maior valor digitado é: ",MAIOR);  
fim
```



```
program l3p070;
var
  VALOR, MENOR, MAIOR: real;
begin
  write('Digite um valor: ');
  readLn(VALOR);
  if ( VALOR <> -1 ) then begin
    MENOR := VALOR;
    MAIOR := VALOR;  end
  else begin
    MENOR := 0;
    MAIOR := 0;  end;
  while ( VALOR <> -1 ) do
    begin
      write('Digite um valor: ');
      readLn(VALOR);
      if ( VALOR > 0 ) then
        if ( VALOR < MENOR ) then
          MENOR := VALOR
        else if ( VALOR > MAIOR ) then
          MAIOR := VALOR;
      end;
      writeLn('O menor valor digitado eh: ',MENOR);
      writeLn('O maior valor digitado eh: ',MAIOR);
    end.
end.
```

```
program l3p070b;
var
  VALOR, MENOR, MAIOR: real;
  PRI: boolean;
begin
  PRI := true;
  MENOR := 0;
  MAIOR := 0;
  repeat
    write('Digite um valor: ');
    readLn(VALOR);
    if ( VALOR > 0 ) then
      begin
        if ( VALOR < MENOR ) or ( PRI ) then
          MENOR := VALOR;
        if ( VALOR > MAIOR ) or ( PRI ) then
          MAIOR := VALOR;
        PRI := false;
      end;
  until ( VALOR = -1 );
  writeLn('O menor valor digitado eh: ',MENOR);
  writeLn('O maior valor digitado eh: ',MAIOR);
end.
```



```
VALOR = input('Digite um valor: ');
if ( VALOR ~= -1 )
    MENOR = VALOR;
    MAIOR = VALOR;
else
    MENOR = VALOR;
    MAIOR = VALOR;
end
while ( VALOR ~= -1 )
    VALOR = input('Digite um valor: ');
    if ( VALOR > 0 )
        if ( VALOR < MENOR )
            MENOR = VALOR;
        else
            if ( VALOR > MAIOR )
                MAIOR = VALOR;
            end
        end
    end
end
end
fprintf(1,'O menor valor digitado eh: %f\n',MENOR);
fprintf(1,'O maior valor digitado eh: %f\n',MAIOR);
```



- 71) Dado um país A, com 5000000 de habitantes e uma taxa de natalidade de 3% ao ano, e um país B com 7000000 de habitantes e uma taxa de natalidade de 2% ao ano, calcular e imprimir o tempo necessário para que a população do país A ultrapasse a população do país B.

```
algoritmo L3P071;  
var  
  inteiro: T;  
  real: A, B;  
início  
  A ← 5;  
  B ← 7;  
  T ← 0;  
  enquanto ( A < B ) faça  
    A ← A + A * 3/100;  
    B ← B + B * 2/100;  
    T ← T + 1; { passaram-se um ano }  
  fim-enquanto  
  imprima("Foram necessários ",T," anos para que a população A ultrapassasse B");  
fim
```

```
program l3p071;  
var  
  T: integer;  
  A, B: real;  
begin  
  A := 5;  
  B := 7;  
  T := 0;  
  while ( A < B ) do  
    begin  
      A := A + A * 3/100;  
      B := B + B * 2/100;  
      T := T + 1; { passaram-se um ano }  
    end;  
  writeln('Foram necessarios ',T,' anos para que a populacao A ultrapassasse B');  
end.
```

```
%T: integer;  
%A, B: real;  
A = 5;  
B = 7;  
T = 0;  
while ( A < B )  
  A = A + A * 3/100;  
  B = B + B * 2/100;  
  T = T + 1; % passaram-se um ano  
end  
fprintf(1,'Foram necessarios %d anos para que a populacao A ultrapassasse B\n',T);
```



- 72) Chico tem 1,50m e cresce 2 centímetros por ano, enquanto Juca tem 1,10m e cresce 3 centímetros por ano. Construir um algoritmo em PORTUGOL que calcule e imprima quantos anos serão necessários para que Juca seja maior que Chico.

```
algoritmo L3P072;  
var  
  inteiro: T;  
  real: CHICO, JUCA;  
início  
  CHICO ← 1.5;  
  JUCA ← 1.1;  
  T ← 0;  
  enquanto ( CHICO < JUCA ) faça  
    CHICO ← CHICO + 0.02;  
    JUCA ← JUCA + 0.03;  
    T ← T + 1; { passaram-se um ano }  
  fim-enquanto  
  imprima("Serão necessários ",T," anos para que Juca fique maior que Chico");  
fim  
  
program l3p072;  
var  
  T: integer;  
  CHICO, JUCA: real;  
begin  
  CHICO := 1.5;  
  JUCA := 1.1;  
  T := 0;  
  while ( JUCA < CHICO ) do  
    begin  
      CHICO := CHICO + 0.02;  
      JUCA := JUCA + 0.03;  
      T := T + 1; { passaram-se um ano }  
    end;  
  writeln('Serao necessarios ',T,' anos para que Juca fique maior que Chico');  
end.  
  
% T: integer;  
%CHICO, JUCA: real;  
CHICO = 1.5;  
JUCA = 1.1;  
T = 0;  
while ( JUCA < CHICO )  
  CHICO = CHICO + 0.02;  
  JUCA = JUCA + 0.03;  
  T = T + 1; % passaram-se um ano  
end  
fprintf(1,'Serao necessarios %d anos para que Juca fique maior que Chico\n',T);
```



- 73) Na usina de Angra dos Reis, os técnicos analisam a perda de massa de um material radioativo. Sabendo-se que este perde 25% de sua massa a cada 30 segundos. Criar um algoritmo em PORTUGOL que imprima o tempo necessário para que a massa deste material se torne menor que 0,10 grama. O algoritmo pode calcular o tempo para várias massas.

```
algoritmo L3P073;  
var  
  inteiro: T;  
  real: M;  
inicio  
  leia(M);  
  enquanto (M > 0) faça  
    T ← 0;  
    enquanto ( M >= 0.1 ) faça  
      M ← M - ( 25 / 100 ) * M;  
      T ← T + 30; { passaram-se 30 segundos }  
    fim-enquanto  
    imprima ("Serão necessários ",T," segundos");  
    leia(M);  
  fim-enquanto  
fim
```

```
program l3p073;  
var  
  T: integer;  
  M: real;  
begin  
  write('Digite a massa: ');  
  readLn(M);  
  T := 0;  
  while ( M >= 0.1 ) do  
    begin  
      M := M - ( 25 / 100 ) * M;  
      T := T + 30; { passaram-se 30 segundos }  
    end;  
  writeLn('Serao necessarios ',T,' segundos');  
end.
```

```
% T: integer;  
% M: real;  
M = input('Digite a massa: ');  
T = 0;  
while ( M >= 0.1 )  
  M = M - ( 25 / 100 ) * M;  
  T = T + 30; % passaram-se 30 segundos  
end  
fprintf(1,'Serao necessarios %d segundos\n',T);
```



- 74) Dois ciclistas A e B estão andando em uma pista de ciclismo com 2 Km de comprimento com velocidades de 10 m/s e 15 m/s, respectivamente. Escreva um algoritmo em PORTUGOL que determine o tempo que levará para que esses dois ciclistas A e B se encontrem em um mesmo ponto, sabendo que eles partiram de um mesmo ponto inicial, porém em sentido contrário. O algoritmo também deve calcular o deslocamento (a distância) que cada um percorreu.

```
algoritmo L3P074;
var
  inteiro: T;
  real: A, B;
início
  A ← 0;
  B ← 2000;
  T ← 0;
  enquanto ( A < B ) faça
    A ← A + 10;
    B ← B - 15;
    T ← T + 1; { passaram-se um segundo }
  fim-enquanto
  imprima("Foram necessários ",T," segundos para o encontro ");
  imprima("A ciclista A percorreu: ",A," metros ");
  imprima("Enquanto que o ciclista B percorreu: ",2000-B," metros ");
fim

program l3p074;
var
  T: integer;
  A, B: real;
begin
  A := 0;
  B := 2000;
  T := 0;
  while ( A < B ) do
    begin
      A := A + 10;
      B := B - 15;
      T := T + 1; { passaram-if um segundo }
    end;
  writeln('Foram necessarios ',T,' segundos para o encontro ');
  writeln('O ciclista A percorreu: ',A:1:0,' metros ');
  writeln('Enquanto que o ciclista B percorreu: ',2000-B:1:0,' metros ');
end.

% T: integer;
% A, B: real;
A = 0;
B = 2000;
T = 0;
while ( A < B )
  A = A + 10;
  B = B - 15;
  T = T + 1; % passaram-if um segundo
end
fprintf(1,'Foram necessarios %d segundos para o encontro\n',T);
fprintf(1,'O ciclista A percorreu: %f metros\n',A);
fprintf(1,'Enquanto que o ciclista B percorreu: %f metros\n',2000-B);
```




- 75) Considere uma linha ferroviária entre São Paulo e Curitiba. Suponha que uma locomotiva (trem) A parte de São Paulo para Curitiba com velocidade de 30 m/s enquanto que uma outra locomotiva B parte de Curitiba para São Paulo no mesmo instante com velocidade de 40 m/s. Considere a distância entre São Paulo e Curitiba de 400 Km. Escreva um algoritmo em PORTUGOL que calcule o tempo necessário para os maquinistas pararem as locomotivas antes que uma colisão aconteça. O algoritmo deve calcular também a distância que as locomotivas devem percorrer para que a colisão aconteça.

```
algoritmo L3P075;
var
  inteiro: T;
  real: A, B;
início
  A ← 0;
  B ← 400000;
  T ← 0;
  enquanto ( A < B ) faça
    A ← A + 30;
    B ← B - 40;
    T ← T + 1; { passaram-se um segundo }
  fim-enquanto
  imprima("São necessários ",T," segundos para a colisão");
  imprima("A locomotiva A deve percorrer: ",A," metros");
  imprima("Enquanto que a locomotiva B deve percorrer: ",400000-B," metros");
fim

program l3p075;
var
  T: integer;
  A, B: real;
begin
  A := 0;
  B := 400000;
  T := 0;
  while ( A < B ) do
    begin
      A := A + 30;
      B := B - 40;
      T := T + 1; { passaram-if um segundo }
    end;
  writeln('Sao necessarios ',T,' segundos para a colisao');
  writeln('A locomotiva A deve percorrer: ',A:1:0,' metros');
  writeln('Enquanto que a locomotiva B deve percorrer: ',400000-B:1:0,' metros');
end.

A = 0;
B = 400000;
T = 0;
while ( A < B )
  A = A + 30;
  B = B - 40;
  T = T + 1; % passaram-se um segundo
end
fprintf(1,'Sao necessarios %d segundos para a colisao\n',T);
fprintf(1,'A locomotiva A deve percorrer: %f metros\n',A);
fprintf(1,'Enquanto que a locomotiva B deve percorrer: %f metros\n',400000-B);
```



76) Criar um algoritmo em PORTUGOL que receba vários números inteiros e positivos e imprima o produto dos números ímpares digitados e a soma dos pares. O algoritmo encerra quando o zero ou um número negativo é digitado.

```
algoritmo L3P076;  
var  
  inteiro: VALOR, PRODPAR, SOMAIMP;  
inicio  
  PRODPAR ← 1;  
  SOMAIMP ← 0;  
  leia(VALOR);  
  enquanto ( VALOR > 0 ) faça  
    se ( VALOR mod 2 = 0 ) { é par? } então  
      PRODPAR ← PRODPAR * VALOR;  
    senão  
      SOMAIMP ← SOMAIMP + VALOR;  
    fim-se  
  leia(VALOR);  
fim-enquanto  
imprima("O produto dos valores pares digitados é: ",PRODPAR);  
imprima("A soma dos valores ímpares digitados é: ",SOMAIMP);  
fim
```

```
algoritmo L3P076B;  
var  
  inteiro: VALOR, PRODPAR, SOMAIMP;  
inicio  
  PRODPAR ← 1;  
  SOMAIMP ← 0;  
  faça  
    leia(VALOR);  
    se ( VALOR > 0 ) então  
      se ( VALOR mod 2 = 0 ) { é par? } então  
        PRODPAR ← PRODPAR * VALOR;  
      senão  
        SOMAIMP ← SOMAIMP + VALOR;  
      fim-se  
    fim-se  
  enquanto ( VALOR > 0 );  
imprima("O produto dos valores pares digitados é: ",PRODPAR);  
imprima("A soma dos valores ímpares digitados é: ",SOMAIMP);  
fim
```

```
program l3p076;  
var  
  PRODPAR: real;  
  VALOR, SOMAIMP: integer;  
begin  
  PRODPAR := 1;  
  SOMAIMP := 0;  
  write('Digite um valor: ');  
  readLn(VALOR);  
  while ( VALOR > 0 ) do  
    begin  
      if ( VALOR mod 2 = 0 ) then { e par<- }  
        PRODPAR := PRODPAR * VALOR  
      else  
        SOMAIMP := SOMAIMP + VALOR;  
        write('Digite um valor: ');  
        readLn(VALOR);  
      end;  
    writeLn('O produto dos valores pares digitados eh: ',PRODPAR:1:0);  
    writeLn('A soma dos valores impares digitados eh: ',SOMAIMP);  
  end.
```



```
program l3p076b;
var
  PRODPAR: real;
  VALOR, SOMAIMP: integer;
begin
  PRODPAR := 1;
  SOMAIMP := 0;
  repeat
    write('Digite um valor: ');
    readLn(VALOR);
    if ( VALOR > 0 ) then
      if ( VALOR mod 2 = 0 ) { é par<- } then
        PRODPAR := PRODPAR * VALOR
      else
        SOMAIMP := SOMAIMP + VALOR;
    until ( VALOR <= 0 );
    writeLn('O produto dos valores pares digitados eh: ',PRODPAR:1:0);
    writeLn('A soma dos valores impares digitados e: ',SOMAIMP);
end.

% PRODPAR: real;
% VALOR, SOMAIMP: integer;
PRODPAR = 1;
SOMAIMP = 0;
VALOR = input('Digite um valor: ');
while ( VALOR > 0 )
  if ( mod(VALOR,2) == 0 ) % e par
    PRODPAR = PRODPAR * VALOR;
  else
    SOMAIMP = SOMAIMP + VALOR;
  end
  VALOR = input('Digite um valor: ');
end
fprintf(1,'O produto dos valores pares digitados eh: %d\n',PRODPAR);
fprintf(1,'A soma dos valores impares digitados eh: %d\n',SOMAIMP);
```



77) Criar um algoritmo em PORTUGOL que leia vários números terminados por 0 e imprima o maior, o menor e a média aritmética dos números. O número 0 (zero) não faz parte da seqüência.

```
algoritmo L3P077;  
var  
  real: NUM, MAIOR, MENOR, MEDIA, SOMA;  
  inteiro: C;  
início  
  leia(NUM);  
  MAIOR ← NUM;  
  MENOR ← NUM;  
  SOMA ← NUM;  
  C ← 1;  
  enquanto ( NUM ≠ 0 ) faça  
    leia(NUM);  
    se (NUM mod 10 = 0 e NUM ≠ 0) então  
      se ( NUM > MAIOR ) então  
        MAIOR ← NUM;  
      fim-se  
      se ( NUM < MENOR ) então  
        MENOR ← NUM;  
      fim-se  
      SOMA ← SOMA + NUM;  
      C ← C + 1;  
    fim-se  
  fim-enquanto  
  MEDIA ← SOMA / C;  
  imprima("O maior valor digitado é: ",MAIOR);  
  imprima("O menor valor digitado é: ",MENOR);  
  imprima("A média dos valores digitados é: ",MEDIA);  
fim
```

```
program l3p077;  
var  
  MAIOR,MENOR,MEDIA,SOMA: real;  
  NUM, C: integer;  
begin  
  write('Digite um numero: ');  
  readLn(NUM);  
  MAIOR := NUM;  
  MENOR := NUM;  
  SOMA := NUM;  
  C := 1;  
  while ( NUM <> 0 ) do  
    begin  
      write('Digite um numero: ');  
      readLn(NUM);  
      if ((NUM mod 10 = 0) AND (NUM <> 0)) then  
        begin  
          if ( NUM > MAIOR ) then  
            MAIOR := NUM;  
          if ( NUM < MENOR ) then  
            MENOR := NUM;  
          SOMA := SOMA + NUM;  
          C := C + 1;  
        end;  
      end;  
    MEDIA := SOMA / C;  
    writeLn('O maior valor digitado eh: ',MAIOR:5:4);  
    writeLn('O menor valor digitado eh: ',MENOR:5:4);  
    writeLn('A media dos valores digitados eh: ',MEDIA:5:4);  
  end.
```



```
% NUM, MAIOR, MENOR, MEDIA, SOMA: real;
% C: integer;
NUM = input('Digite um numero: ');
MAIOR = NUM;
MENOR = NUM;
SOMA = NUM;
C = 1;
while ( NUM ~= 0 )
    NUM = input('Digite um numero: ');
    if (mod(NUM,10) == 0 & NUM ~= 0 )
        if ( NUM > MAIOR )
            MAIOR = NUM;
        end
        if ( NUM < MENOR )
            MENOR = NUM;
        end
        SOMA = SOMA + NUM;
        C = C + 1;
    end
end
MEDIA = SOMA / C;
fprintf(1,'O maior valor digitado eh: %f\n',MAIOR);
fprintf(1,'O menor valor digitado eh: %f\n',MENOR);
fprintf(1,'A media dos valores digitados eh: %f\n',MEDIA);
```



- 78) Criar um algoritmo em PORTUGOL que leia idade e sexo (0-masculino, 1-feminino) de várias pessoas. Calcule e imprima a idade média, o total de pessoas do sexo feminino com idade entre 30-45 inclusive e o número total de pessoas do sexo masculino. O algoritmo termina quando se digita um número não positivo (0 ou um número negativo) para a idade.

```
algoritmo L3P078;  
var  
  real: MEDIA, SOMA, TOT3045, TOTMAS;  
  inteiro: IDADE, SEXO, C;  
inicio  
  SOMA ← 0;  
  C ← 0;  
  TOT3045 ← 0;  
  TOTMAS ← 0;  
  leia(IDADE)  
  enquanto ( IDADE > 0 ) faça  
    leia(SEXO);  
    SOMA ← SOMA + IDADE;  
    C ← C + 1;  
    se ( SEXO = 1 e IDADE >= 30 e IDADE <= 45 ) então  
      TOT3045 ← TOT3045 + 1;  
    fim-se  
    se (SEXO = 0) então  
      TOTMAS ← TOTMAS + 1;  
    fim-se  
  leia(IDADE)  
fim-enquanto  
  MEDIA ← SOMA / C;  
  imprima("A média de idade é: ",MEDIA);  
  imprima("O total de mulheres com idade no intervalo [30,45] é: ",TOT3045);  
  imprima("O total de pessoas do sexo masculino é: ",TOTMAS);  
fim
```

```
program l3p078;  
var  
  MEDIA, SOMA: real;  
  IDADE, SEXO, C, TOT3045, TOTMAS: integer;  
begin  
  SOMA := 0;  
  C := 0;  
  TOT3045 := 0;  
  TOTMAS := 0;  
  write('Digite a idade: ');  
  readLn(IDADE);  
  while ( IDADE > 0 ) do  
    begin  
      write('Digite o sexo: ');  
      readLn(SEXO);  
      SOMA := SOMA + IDADE;  
      C := C + 1;  
      if ( SEXO = 1 ) and ( IDADE >= 30 ) and ( IDADE <= 45 ) then  
        TOT3045 := TOT3045 + 1;  
      if (SEXO = 0) then  
        TOTMAS := TOTMAS + 1;  
      write('Digite a idade: ');  
      read(IDADE)  
    end;  
  MEDIA := SOMA / C;  
  writeLn('A media de idade eh: ',MEDIA:3:1);  
  writeLn('O total de mulheres com idade no intervalo [30,45] eh: ',TOT3045);  
  writeLn('O total de pessoas do sexo masculino eh: ',TOTMAS);  
end.
```



```
% MEDIA, SOMA: real;
% IDADE, SEXO, C, TOT3045, TOTMAS: integer;
SOMA = 0;
C = 0;
TOT3045 = 0;
TOTMAS = 0;
IDADE = input('Digite a idade: ');
while ( IDADE > 0 )
    SEXO = input('Digite o sexo: ');
    SOMA = SOMA + IDADE;
    C = C + 1;
    if ( SEXO == 1 ) & ( IDADE >= 30 ) & ( IDADE <= 45 )
        TOT3045 = TOT3045 + 1;
    end
    if (SEXO == 0)
        TOTMAS = TOTMAS + 1;
    end
    IDADE = input('Digite a idade: ');
end
MEDIA = SOMA / C;
fprintf(1,'A media de idade eh: %f\n',MEDIA);
fprintf(1,'O total de mulheres com idade no intervalo [30,45] eh: %d\n',TOT3045);
fprintf(1,'O total de pessoas do sexo masculino eh: %d\n',TOTMAS);
```



79) Os alunos de informática tiveram cinco provas (uma de cada matéria/disciplina): 1, 2, 3, 4 e 5. Considere aprovado o aluno com nota igual ou superior a 7.0. Criar um algoritmo em PORTUGOL que leia o Nome dos alunos e as suas respectivas cinco notas. Inicialmente o algoritmo deve perguntar quantos alunos existem na turma e deve também imprimir:

- Nome dos que foram aprovados em todas as matérias;
- Nome dos alunos aprovados nas matérias 1 e 4;
- A porcentagem dos aprovados na matéria 3.

```
algoritmo L3P079;
var
  real:    NOTA1, NOTA2, NOTA3, NOTA4, NOTA5, POR3;
  inteiro: CT3;
  literal: NOME;
início
  CT3 ← 0 ;
  leia(NTURMA);
  para C de 1 até NTURMA faça
    leia(NOME);
    leia(NOTA1,NOTA2,NOTA3,NOTA4,NOTA5);
    se ( NOTA1 >= 7 e NOTA2 >= 7 e NOTA3 >= 7 e NOTA4 >= 7 e NOTA5 >= 7 ) então
      imprima(NOME," aprovado em todas as matérias! ");
    senão-se ( NOTA1 >= 7 e NOTA4 >= 7 ) então
      imprima(NOME," aprovado nas matérias 1 e 4! ");
    fim-se
    se ( NOTA3 >= 7 ) então
      CT3 ← CT3 + 1;
    fim-se
  fim-para
  POR3 ← 100* CT3 / NTURMA;
  imprima("A porcentagem dos alunos aprovados na matéria 3 é: ",POR3);
fim
```

```
program l3p079;
var
  NOTA1, NOTA2, NOTA3, NOTA4, NOTA5, POR3: real;
  NTURMA, C, CT3: integer;
  NOME: string;
begin
  CT3 := 0 ;
  write('Quantos alunos existem na sala: ');
  readLN(NTURMA);
  for C := 1 to NTURMA do
    begin
      write('Digite o nome do aluno: ');
      readLn(NOME);
      write('Digite as 5 notas: ');
      readLn(NOTA1,NOTA2,NOTA3,NOTA4,NOTA5);
      if ( NOTA1 >= 7 ) and ( NOTA2 >= 7 ) and ( NOTA3 >= 7 ) and ( NOTA4 >= 7 ) and ( NOTA5 >=
7 ) then
        writeln(NOME,' aprovado em todas as materias! ')
      else if ( NOTA1 >= 7 ) and ( NOTA4 >= 7 ) then
        writeln(NOME,' aprovado nas materias 1 e 4! ');
      if ( NOTA3 >= 7 ) then
        CT3 := CT3 + 1;
      end;
    POR3 := 100* CT3 / NTURMA;
    writeln('A porcentagem dos alunos aprovados na materia 3 eh: ',POR3:6:2);
  end.
```




```
% NOTA1, NOTA2, NOTA3, NOTA4, NOTA5, POR3: real;
% NTURMA, C, CT3: integer;
% NOME: string;
CT3 = 0 ;
NTURMA = input('Quantos alunos existem na sala: ');
for C = 1 : NTURMA
    NOME = input('Digite o nome do aluno: ','s');
    NOTA1 = input('Digite a 1a. nota: ');
    NOTA2 = input('Digite a 2a. nota: ');
    NOTA3 = input('Digite a 3a. nota: ');
    NOTA4 = input('Digite a 4a. nota: ');
    NOTA5 = input('Digite a 5a. nota: ');
    if ( NOTA1 >= 7 ) & ( NOTA2 >= 7 ) & ( NOTA3 >= 7 ) & ( NOTA4 >= 7 ) & ( NOTA5 >= 7 )
        fprintf(1,'%s aprovado em todas as materias!\n',NOME);
    elseif ( NOTA1 >= 7 ) & ( NOTA4 >= 7 )
        fprintf(1,'%s aprovado nas materias 1 e 4!\n',NOME);
    end
    if ( NOTA3 >= 7 )
        CT3 = CT3 + 1;
    end
end
POR3 = 100* CT3 / NTURMA;
fprintf(1,'A porcentagem dos alunos aprovados na materia 3 eh: %.2f\n',POR3);
```



80) Criar um algoritmo em PORTUGOL que leia um conjunto de informações (nome, sexo, idade, peso e altura) dos atletas que participaram de uma olimpíada, e informar:

- O atleta do sexo masculino mais alto;
- A atleta do sexo feminino mais pesada;
- A média de idade dos atletas.

Deverão ser lidos dados dos atletas até que seja digitado o nome @ para um atleta.

```
algoritmo L3P080;
var
  real: PESO, ALTURA, ALTUMAS, PESOFEM, MEDIAIDADE;
  inteiro: IDADE, SOMA, CTMAS, CTFEM;
  literal: NOME, SEXO;
início
  CTMAS ← 0;
  CTFEM ← 0;
  SOMA ← 0;
  ALTUMAS ← 0;
  PESOFEM ← 0;
  leia(NOME);
  enquanto ( NOME ≠ "@" ) faça
    leia(SEXO, PESO, ALTURA, IDADE)
    se ( SEXO = "M" ) então
      CTMAS ← CTMAS + 1;
      se ( ALTURA > ALTUMAS ou CTMAS = 1 ) então
        ALTUMAS ← ALTURA;
      fim-se
    fim-se
    se ( SEXO = "F" ) então
      CTFEM ← CTFEM + 1;
      se ( PESO > PESOFEM ou CTFEM = 1 ) então
        PESOFEM ← PESO;
      fim-se
    fim-se
    SOMA ← SOMA + IDADE;
    leia(NOME);
  fim-enquanto
  MEDIA ← SOMA / (CTMAS + CTFEM);
  imprima("A altura do mais alto atleta do sexo masculino é: ", ALTUMAS);
  imprima("O peso da atleta feminina mais pesada é: ", PESOFEM);
  imprima("A média de idade dos atletas é: ", MEDIA);
fim
```



```
program l3p080;
var
  PESO,ALTURA,ALTUMAS,PESOFEM, MEDIA: real;
  IDADE,SOMA,CTMAS,CTFEM: integer;
  SEXO, NOME: string;
begin
  CTMAS := 0;
  CTFEM := 0;
  SOMA := 0;
  ALTUMAS := 0;
  PESOFEM := 0;
  write('Digite o nome: ');
  readLn(NOME);
  while ( NOME <> '@' ) do
  begin
    write('Sexo (M/F): ');
    readLn(SEXO);
    write('Peso: ');
    readLn(PESO);
    write('Altura: ');
    readLn(ALTURA);
    write('Idade: ');
    readLn(IDADE);
    if ( SEXO = 'M' ) or ( SEXO = 'm' ) then
    begin
      CTMAS := CTMAS + 1;
      if ( ALTURA > ALTUMAS ) or ( CTMAS = 1 ) then
        ALTUMAS := ALTURA;
    end;
    if ( SEXO = 'F' ) or ( SEXO = 'f' ) then
    begin
      CTFEM := CTFEM + 1;
      if ( PESO > PESOFEM ) or ( CTFEM = 1 ) then
        PESOFEM := PESO;
    end;
    SOMA := SOMA + IDADE;
    write('Digite o nome: ');
    readLn(NOME);
  end;
  writeLn('A altura do mais alto atleta do sexo masculino eh: ',ALTUMAS:4:2);
  writeLn('O peso da atleta feminina mais pesada eh: ',PESOFEM:5:2);
  if ( CTMAS + CTFEM <> 0 ) then begin
    MEDIA := SOMA / (CTMAS + CTFEM);
    writeLn('A media de idade dos atletas eh: ',MEDIA:3:2); end
  else
    writeLn('Nenhum atleta foi detectado!');
end.
```



```
CTMAS = 0;
CTFEM = 0;
SOMA = 0;
ALTUMAS = 0;
PESOFEM = 0;
NOME = input('Digite o nome: ');
while ( NOME ~= '@' )
    SEXO = input('Sexo (M/F): ');
    PESO = input('Peso: ');
    ALTURA = input('Altura: ');
    IDADE = input('Idade: ');
    if ( SEXO == 'M' ) | ( SEXO == 'm' )
        CTMAS = CTMAS + 1;
        if ( ALTURA > ALTUMAS ) | ( CTMAS == 1 )
            ALTUMAS = ALTURA;
        end
    end
    if ( SEXO == 'F' ) | ( SEXO == 'f' )
        CTFEM = CTFEM + 1;
        if ( PESO > PESOFEM ) | ( CTFEM == 1 )
            PESOFEM = PESO;
        end
    end
    SOMA = SOMA + IDADE;
    NOME = input('Digite o nome: ');
end
fprintf(1,'A altura do mais alto atleta do sexo masculino eh: %.2f\n',ALTUMAS);
fprintf(1,'O peso da atleta feminina mais pesada eh: %.2f\n',PESOFEM);
if ( CTMAS + CTFEM ~= 0 )
    MEDIA = SOMA / (CTMAS + CTFEM);
    fprintf(1,'A media de idade dos atletas eh: %.2f\n',MEDIA);
else
    disp('Nenhum atleta foi detectado!');
end
```



81) Escreva um algoritmo em PORTUGOL que receba a idade de várias pessoas e imprima:

- O total de pessoas com menos de 21 anos;
- O total de pessoas com mais de 50 anos.

```
algoritmo L3P081;  
var  
  inteiro: IDADE, I21, I50;  
início  
  I21 ← 0;  
  I50 ← 0;  
  leia(IDADE);  
  enquanto ( IDADE > 0 ) faça  
    se ( IDADE < 21 ) então  
      I21 ← I21 + 1;  
    fim-se  
    se ( IDADE > 50 ) então  
      I50 ← I50 + 1;  
    fim-se  
  leia(IDADE);  
  fim-enquanto  
  imprima("O total de pessoas com menos de 21 anos é: ", I21);  
  imprima("O total de pessoas com mais de 50 anos é: ", I50);  
fim
```

```
algoritmo L3P081B;  
var  
  inteiro: IDADE, I21, I50;  
início  
  I21 ← 0;  
  I50 ← 0;  
  repita:  
    leia(IDADE);  
    se ( IDADE > 0 ) então  
      se ( IDADE < 21 ) então  
        I21 ← I21 + 1;  
      senão-se ( IDADE > 50 ) então  
        I50 ← I50 + 1;  
      fim-se  
    fim-se  
  enquanto ( IDADE > 0 );  
  imprima("O total de pessoas com menos de 21 anos é: ", I21);  
  imprima("O total de pessoas com mais de 50 anos é: ", I50);  
fim
```

```
program l3p081;  
var  
  IDADE, I21, I50: integer;  
begin  
  I21 := 0;  
  I50 := 0;  
  write('Digite uma idade: ');  
  readLn(IDADE);  
  while ( IDADE > 0 ) do  
    begin  
      if ( IDADE < 21 ) then  
        I21 := I21 + 1;  
      if ( IDADE > 50 ) then  
        I50 := I50 + 1;  
      write('Digite uma idade: ');  
      readLn(IDADE);  
    end;  
  writeLn('O total de pessoas com menos de 21 anos eh: ', I21);  
  writeLn('O total de pessoas com mais de 50 anos eh: ', I50);  
end.
```



```
program l3p081b;
var
  IDADE, I21,I50: integer;
begin
  I21 := 0;
  I50 := 0;
  repeat
    write('Digite uma idade: ');
    readLn(IDADE);
    if ( IDADE > 0 ) then
      if ( IDADE < 21 ) then
        I21 := I21 + 1
      else if ( IDADE > 50 ) then
        I50 := I50 + 1;
    until ( IDADE <= 0 );
    writeLn('O total de pessoas com menos de 21 anos eh: ',I21);
    writeLn('O total de pessoas com mais de 50 anos eh: ' ,I50);
end.

% IDADE, I21,I50: integer;
I21 = 0;
I50 = 0;
IDADE = input('Digite uma idade: ');
while ( IDADE > 0 )
  if ( IDADE < 21 )
    I21 = I21 + 1;
  end
  if ( IDADE > 50 )
    I50 = I50 + 1;
  end
  IDADE = input('Digite uma idade: ');
end
fprintf(1,'O total de pessoas com menos de 21 anos eh: %d\n',I21);
fprintf(1,'O total de pessoas com mais de 50 anos eh: %d\n' ,I50);
```



82) Construa um algoritmo em PORTUGOL que receba um número e verifique se ele é um número triangular. (Um número é triangular quando é resultado do produto de três números consecutivos. Exemplo: $24 = 2 \times 3 \times 4$)

```
algoritmo L3P082;  
var  
  inteiro: C, RESULT, NUM;  
inicio  
  C ← 1;  
  leia(NUM);  
  repita:  
    RESULT ← C * ( C + 1 ) * ( C + 2 ) ;  
    se ( RESULT = NUM ) então  
      imprima("o número é triangular");  
    fim-se  
    C ← C + 1;  
  enquanto ( RESULT >= NUM );  
fim
```

```
program l3p082;  
var  
  C, RESULT, NUM: integer;  
begin  
  C := 1;  
  write('Digite um numero: ');  
  readLn(NUM);  
  repeat  
    RESULT := C * ( C + 1 ) * ( C + 2 ) ;  
    if ( RESULT = NUM ) then  
      writeLn('o numero eh triangular');  
    C := C + 1;  
  until ( RESULT >= NUM);  
end.
```

```
% C, RESULT, NUM: integer;  
C = 1;  
NUM = input('Digite um numero: ');  
RESULT = C * ( C + 1 ) * ( C + 2 ) ;  
while ( RESULT <= NUM )  
  if ( RESULT == NUM )  
    fprintf(1,'o numero eh triangular');  
  end  
  C = C + 1;  
  RESULT = C * ( C + 1 ) * ( C + 2 ) ;  
end
```



83) Escreva um algoritmo em PORTUGOL que receba vários números e ao final imprima o maior número digitado. O algoritmo acaba quando se digita -9999;

```
algoritmo L3P083;  
var  
  real: VALOR, MAIOR;  
início  
  leia(VALOR);  
  MAIOR ← VALOR  
  enquanto ( VALOR ≠ -9999 ) faça  
    se ( VALOR > MAIOR ) então  
      MAIOR ← VALOR;  
    fim-se  
  leia(VALOR);  
fim-enquanto  
  imprima("O maior valor digitado é: ",MAIOR);  
fim  
  
program l3p083;  
var  
  VALOR, MAIOR: real;  
begin  
  write('Digite o valor: ');  
  readLn(VALOR);  
  MAIOR := VALOR;  
  while ( VALOR <> -9999 ) do  
    begin  
      if ( VALOR > MAIOR ) then  
        MAIOR := VALOR;  
      write('Digite o valor: ');  
      readLn(VALOR);  
    end;  
    writeLn('O maior valor digitado eh: ',MAIOR);  
  end.  
  
% VALOR, MAIOR: integer;  
VALOR = input('Digite o valor: ');  
MAIOR = VALOR;  
while ( VALOR ~= -9999 )  
  if ( VALOR > MAIOR )  
    MAIOR = VALOR;  
  end  
  VALOR = input('Digite o valor: ');  
end  
fprintf(1,'O maior valor digitado eh: %f\n',MAIOR);
```




84) Criar um algoritmo em PORTUGO que calcule o M.M.C (mínimo múltiplo comum) entre dois números lidos. (por exemplo: o M.M.C, entre 10 e 15 é 30).

```
algoritmo L3P084;  
var  
  inteiro: A, B, C, MMC;  
início  
  MMC ← 1;  
  leia(A,B);  
  C ← 2;  
  enquanto ( C <= A ou C <= B ) faça  
    se ( A mod C = 0 ou B mod C = 0 ) então  
      MMC ← MMC * C;  
      se ( A mod C = 0 ) então  
        A ← A / C;  
      fim-se  
      se ( B mod C = 0 ) então  
        B ← B / C;  
      fim-se  
      C ← C - 1;  
    fim-se  
    C ← C + 1;  
  fim-enquanto  
  imprima("MMC é: ",MMC);  
fim  
  
program l3p084;  
var  
  A, B, C, MMC: integer;  
begin  
  MMC := 1;  
  write('Digite um numero: ');  
  readLn(A);  
  write('Digite outro numero: ');  
  readLn(B);  
  C := 2;  
  while ( C <= A ) or ( C <= B ) do  
  begin  
    if ( A mod C = 0 ) or ( B mod C = 0 ) then  
    begin  
      MMC := MMC * C;  
      if ( A mod C = 0 ) then  
        A := A div C;  
      if ( B mod C = 0 ) then  
        B := B div C;  
      C := C - 1;  
    end;  
    C := C + 1;  
  end;  
  writeLn('MMC eh: ',MMC);  
end.
```



```
% A, B, C, MMC: integer;
MMC = 1;
A = input('Digite um numero: ');
B = input('Digite outro numero: ');
C = 2;
while (( C <= A ) | ( C <= B ))
    if (( mod(A,C) == 0 ) | ( mod(B,C) == 0 ))
        MMC = MMC * C;
        if ( mod(A,C) == 0 )
            A = A / C;
        end
        if ( mod(B,C) == 0 )
            B = B / C;
        end
        C = C - 1;
    end
    C = C + 1;
end
fprintf(1,'MMC eh: %d\n',MMC);
```



- 85) Criar um algoritmo em PORTUGOL que receba vários números inteiros e positivos e imprima a média dos números múltiplos de 3. . A execução deve encerrar quando um número não positivo for lido.

```
algoritmo L3P085;
var
  inteiro: NUM, SOMA, C;
inicio
  C ← 0;
  SOMA ← 0;
  leia(NUM);
  enquanto ( NUM > 0 ) faça
    se ( NUM mod 3 = 0 ) então
      C ← C + 1;
      SOMA ← SOMA + NUM;
    fim-se
  leia(NUM);
fim-enquanto
  se ( C = 0 ) então
    MEDIA ← SOMA / C;
    imprima("A média dos múltiplos de 3 digitados é: ",MEDIA);
  senão
    imprima("Impossivel calcular media!");
  fim-se
fim

program l3p085;
var
  MEDIA: real;
  NUM, SOMA, C: integer;
begin
  C := 0;
  SOMA := 0;
  write('Digite um numero: ');
  readLn(NUM);
  while ( NUM > 0 ) do
    begin
      if ( NUM mod 3 = 0 ) then begin
        C := C + 1;
        SOMA := SOMA + NUM; end;
      write('Digite um numero: ');
      readLn(NUM);
    end;
  if ( C = 0 ) then begin
    MEDIA := SOMA / C;
    writeln('A media dos multiplos de 3 digitados eh: ',MEDIA:5:4); end
  else
    writeln('Impossivel calcular media!');
end.

C = 0;
SOMA = 0;
NUM = input('Digite um numero: ');
while ( NUM > 0 )
  if ( mod(NUM,3) == 0 )
    C = C + 1;
    SOMA = SOMA + NUM;
  end
  NUM = input('Digite um numero: ');
end
if ( C = 0 )
  MEDIA = SOMA / C;
  fprintf(1,'A media dos multiplos de 3 digitados eh: %.4f\n',MEDIA);
else
  disp('Impossível calcular media!');
end
```



- 86) Escreva um algoritmo em PORTUGOL que receba vários números inteiros e imprima a quantidade de números primos dentre os números que foram digitados. O algoritmo acaba quando se digita um número menor ou igual a 0.

```
algoritmo L3P086;  
var  
  inteiro: NUM, NPRIMO, C;  
  lógico: SIT;  
início  
  NPRIMO ← 0;  
  leia(NUM);  
  enquanto ( NUM > 0 ) faça  
    SIT ← verdadeiro;  
    C ← 2  
    enquanto ( SIT e C < NUM ) faça  
      se ( NUM mod C = 0 ) então  
        SIT ← falso;  
      fim-se  
      C ← C + 1;  
    fim-enquanto  
    se ( SIT ) então  
      NPRIMO ← NPRIMO + 1;  
    fim-se  
  leia(NUM);  
  fim-enquanto  
  imprima("O número de números primos que foram digitados é: ", NPRIMO);  
fim
```

```
algoritmo L3P086B;  
var  
  inteiro: NUM, NPRIMO, C;  
  lógico: SIT;  
início  
  NPRIMO ← 0;  
  repita:  
    leia(NUM);  
    se ( NUM > 0 ) então  
      SIT ← verdadeiro;  
      C ← 2;  
      enquanto ( SIT e C < NUM ) faça  
        se ( NUM mod C = 0 ) então  
          SIT ← falso;  
        fim-se  
        C ← C + 1;  
      fim-enquanto  
      se ( SIT ) então  
        NPRIMO ← NPRIMO + 1;  
      fim-se  
    fim-se  
  enquanto ( NUM > 0 );  
  imprima("O número de números primos que foram digitados é: ", NPRIMO);  
fim
```



```
program l3p086;  
var  
  NUM, NPRIMO, C: integer;  
  SIT: boolean;  
begin  
  NPRIMO := 0;  
  write('Digite um numero: ');  
  readLn(NUM);  
  while ( NUM > 0 ) do  
  begin  
    SIT := true;  
    C := 2;  
    while ( SIT ) and ( C < NUM ) do  
    begin  
      if ( NUM mod C = 0 ) then  
        SIT := false;  
      C := C + 1;  
    end;  
    if ( SIT ) then  
      NPRIMO := NPRIMO + 1;  
    write('Digite um numero: ');  
    read(NUM);  
  end;  
  writeLn ('O numero de numeros primos que foram digitados eh: ',NPRIMO);  
end.
```

```
program l3p086b;  
var  
  NUM, NPRIMO, C: integer;  
  SIT: boolean;  
begin  
  NPRIMO := 0;  
  repeat  
    write('Digite um numero: ');  
    readLn(NUM);  
    if ( NUM > 0 ) then  
    begin  
      SIT := true;  
      C := 2;  
      while ( SIT ) and ( C < NUM ) do  
      begin  
        if ( NUM mod C = 0 ) then  
          SIT := false;  
        C := C + 1;  
      end;  
      if ( SIT ) then  
        NPRIMO := NPRIMO + 1;  
    end  
  until ( NUM <= 0 );  
  writeLn('O numero de numeros primos que foram digitados eh: ',NPRIMO);  
end.
```



```
% NUM, NPRIMO, C: integer;
% SIT: boolean;
NPRIMO = 0;
NUM = input('Digite um numero: ');
while ( NUM > 0 )
    SIT = 1; % true
    C = 2;
    while ( SIT ) & ( C < NUM )
        if ( mod(NUM,C) == 0 )
            SIT = 0; % false
        end
        C = C + 1;
    end
    if ( SIT )
        NPRIMO = NPRIMO + 1;
    end
    NUM = input('Digite um numero: ');
end
fprintf(1,'O numero de numeros primos que foram digitados eh: %d\n',NPRIMO);
```



87) Escreva um algoritmo em PORTUGOL que receba vários números, e finalize com a entrada do número -999. Para cada número, o algoritmo deve imprimir seus divisores.

```
algoritmo L3P087;  
var  
  inteiro: NUM, C;  
inicio  
  leia(NUM);  
  enquanto ( NUM ≠ -999 ) faça  
    imprima("São divisores de ",NUM);  
    para C de 2 até NUM faça  
      se ( NUM mod C = 0 ) então  
        imprima(C);  
      fim-se  
    fim-para  
  leia(NUM);  
fim-enquanto  
fim
```

```
program l3p087;  
var  
  NUM, C: integer;  
begin  
  write('Digite um numero: ');  
  readLn(NUM);  
  while ( NUM <> -999 ) do  
    begin  
      writeLn('Sao divisores de ',NUM);  
      for C := 2 to NUM do  
        if ( NUM mod C = 0 ) then  
          write(C, ' ');  
        writeLn('');  
      write('Digite um numero: ');  
      readLn(NUM);  
    end;  
  end.  
  
% NUM, C: integer;  
NUM = input('Digite um numero: ');  
while ( NUM ~= -999 )  
  fprintf(1,'Sao divisores de ',NUM);  
  for C = 2 : NUM  
    if ( mod(NUM,C) == 0 )  
      disp(C);  
    end  
  end  
  NUM = input('Digite um numero: ');  
end
```



- 88) Criar um algoritmo em PORTUGOL que receba 10 números positivos e imprima a raiz quadrada de cada número. Para cada entrada de dados deverá haver um trecho de proteção para que um número negativo não seja aceito.

```
algoritmo L3P088;  
var  
  real : NUM;  
  inteiro : C;  
início  
  para C de 1 até 10 faça  
    leia(NUM);  
    se ( NUM >= 0 ) então  
      imprima( raiz(NUM) );  
    senão  
      imprima("número digitado é negativo! Impossível calcular raiz quadrada!");  
    fim-se  
  fim-para  
fim  
  
program l3p088;  
var  
  NUM : real;  
  C : integer;  
begin  
  for C := 1 to 10 do  
  begin  
    write('Digite um numero: ');  
    readLn(NUM);  
    if ( NUM >= 0 ) then  
      writeLn('Raiz: ' , Sqrt(NUM):5:4 )  
    else  
      writeLn('Numero digitado eh negativo! Impossivel calcular raiz quadrada!');  
    end;  
  end.  
  
% NUM, C: integer;  
for C = 1 : 10  
  NUM = input('Digite um numero: ');  
  if ( NUM >= 0 )  
    fprintf(1,'Raiz: %f\n',sqrt(NUM));  
  else  
    fprintf(1,'Numero digitado eh negativo! Impossivel calcular raiz quadrada!\n');  
  end  
end  
end
```




- 89) Criar um algoritmo em PORTUGOL que leia vários números inteiros e apresente o fatorial de cada número. O algoritmo deve finalizar quando um número menor do que 1 é digitado.

```
algoritmo L3P089;  
var  
  inteiro: NUM, FAT, C;  
inicio  
  leia(NUM);  
  enquanto ( NUM >= 1 ) faça  
    FAT ← 1;  
    para C de 1 até NUM faça  
      FAT ← FAT * C;  
    fim-para  
    imprima(FAT);  
    leia(NUM);  
  fim-enquanto  
fim
```

```
program l3p089;  
var  
  NUM, C: integer;  
  FAT: real;  
begin  
  write('Digite um numero: ');  
  readLn(NUM);  
  while ( NUM >= 1 ) do  
    begin  
      FAT := 1;  
      for C := 1 to NUM do  
        FAT := FAT * C;  
      writeLn(FAT:1:0);  
      write('Digite um numero: ');  
      readLn(NUM);  
    end;  
end.
```

```
NUM = input('Digite um numero: ');  
while ( NUM >= 1 )  
  FAT = 1;  
  for C = 1 : NUM  
    FAT = FAT * C;  
  end  
  disp(FAT);  
  NUM = input('Digite um numero: ');  
end
```



90) Escreva um algoritmo em PORTUGOL que receba vários números e verifique se eles são ou não quadrados perfeitos. O algoritmo termina a execução quando for digitado um número menor ou igual a 0. (Um número é quadrado perfeito quando tem um número inteiro como raiz quadrada.)

```
algoritmo L3P090;  
var  
  inteiro: NUM, C;  
inicio  
  leia(NUM);  
  enquanto ( NUM > 0 ) faça  
    C ← 1;  
    enquanto ( C * C <= NUM) faça  
      se ( C * C = NUM ) então  
        imprima("O número digitado é quadrado perfeito!");  
      fim-se  
      C ← C + 1;  
    fim-enquanto  
  leia(NUM);  
fim-enquanto  
fim
```

```
program l3p090;  
var  
  NUM, C: integer;  
begin  
  write('Digite um numero: ');  
  readLn(NUM);  
  while ( NUM > 0 ) do  
    begin  
      C := 1;  
      while ( C * C <= NUM ) do  
        begin  
          if ( C * C = NUM ) then  
            writeLn('O numero digitado eh quadrado perfeito!');  
          C := C + 1;  
        end;  
      write('Digite um numero: ');  
      readLn(NUM);  
    end;  
  end.  
end.
```

```
NUM = input('Digite um numero: ');  
while ( NUM > 0 )  
  C = 1 ;  
  while ( C * C <= NUM )  
    if ( C * C == NUM )  
      disp('O numero digitado eh quadrado perfeito!');  
    end  
    C = C + 1;  
  end  
NUM = input('Digite um numero: ');  
end
```



91) Numa universidade, os alunos das turmas de informática fizeram uma prova de algoritmos. Cada turma possui um número de alunos. Criar um algoritmo em PORTUGOL que imprima para cada turma:

- Quantidade de alunos aprovados;
- Média de cada turma;
- Percentual de reprovados.

Considere aprovado aluno com nota igual ou superior a 7.0. O programa deve ler o número de alunos de cada turma e só finalizar o programa quando um número não positivo de alunos for digitado.

```
algoritmo L3P091;
var
  inteiro: NTURMA, NALUNO, CT, CA;
  inteiro: NAPR, NREP, NTOT, SOMA;
  real:     NOTA, MEDIA, PERC;
início
  NAPR ← 0;
  NREP ← 0;
  NTOT ← 0;
  imprima("Quantas turmas existem na universidade? ");
  leia(NTURMA);
  para CT de 0 até NTURMA - 1 faça
    SOMA ← 0;
    imprima("Quantos alunos existem nessa turma? ");
    leia(NALUNO);
    para CA de 0 até NALUNO - 1 faça
      leia(NOTA);
      se (NOTA >= 7) então
        NAPR ← NAPR + 1;
      senão
        NREP ← NREP + 1;
    fim-se
    NTOT ← NTOT + 1;
    SOMA ← SOMA + NOTA;
  fim-para
  MEDIA ← SOMA / NALUNO;
  imprima("A média da nota dessa turma é: ", MEDIA);
fim-para
  PERC ← 100 * NREP / NTOT;
  imprima("A quantidade de alunos aprovados é: ", NAPR);
  imprima("O percentual de alunos reprovados é: ", PERC);
fim
```



```
program l3p091;
var
  NTURMA,NALUNO,CT,CA: integer;
  NAPR,NREP,NTOT: integer;
  SOMA,NOTA,MEDIA,PERC: real;
begin
  NAPR := 0;
  NREP := 0;
  NTOT := 0;
  write('Quantas turmas existem na universidade? ');
  read(NTURMA);
  for CT := 0 to NTURMA-1 do
    begin
      SOMA := 0;
      write('Quantos alunos existem nessa turma? ');
      read(NALUNO);
      for CA := 0 to NALUNO-1 do
        begin
          write('Nota: ');
          readLn(NOTA);
          if (NOTA >= 7) then
            NAPR := NAPR + 1
          else
            NREP := NREP + 1;
            NTOT := NTOT + 1;
            SOMA := SOMA + NOTA;
          end;
          MEDIA := SOMA / NALUNO;
          writeln('A media da nota dessa turma eh: ',MEDIA);
        end;
        PERC := 100 * NREP / NTOT;
        writeln('A quantidade de alunos aprovados eh: ',NAPR);
        writeln('O percentual de alunos reprovados eh: ',PERC, '%');
      end.

% NTURMA,NALUNO,CT,CA: integer;
% NAPR,NREP,NTOT: integer;
% SOMA,NOTA,MEDIA,PERC: real;
NAPR = 0;
NREP = 0;
NTOT = 0;
NTURMA = input('Quantas turmas existem na universidade? ');
for CT = 0 : NTURMA-1
  SOMA = 0;
  NALUNO = input('Quantos alunos existem nessa turma? ');
  for CA = 0 : NALUNO-1
    NOTA = input('Nota: ');
    if (NOTA >= 7)
      NAPR = NAPR + 1;
    else
      NREP = NREP + 1;
    end
    NTOT = NTOT + 1;
    SOMA = SOMA + NOTA;
  end
  MEDIA = SOMA / NALUNO;
  fprintf('A media da nota dessa turma eh: %f\n',MEDIA);
end
PERC = 100 * NREP / NTOT;
fprintf('A quantidade de alunos aprovados eh: %d\n',NAPR);
fprintf('O percentual de alunos reprovados eh: %f\n',PERC);
```



92) Numa universidade cada aluno possui os seguintes dados:

- Renda pessoal;
- Renda familiar;
- Total gasto com alimentação;
- Total gasto com outras despesas;

Criar um algoritmo em PORTUGOL que imprima a porcentagem dos alunos que gasta acima de R\$ 200,00 com outras despesas, o número de alunos com renda pessoal maior que renda familiar e a porcentagem gasta com alimentação e outras despesas em relação às rendas pessoal e familiar. O algoritmo acaba quando se digita 0 para a renda pessoal.

```
algoritmo L3P092;
var
  inteiro: NAC200, NTOT, NRDP;
  real: RDP, RDF, TGA, TGO, PAC200;
inicio
  NAC200 ← 0;
  NTOT ← 0;
  NRDP ← 0;
  leia (RDP);
  enquanto ( RDP ≠ 0 ) faça
    leia (RDF);
    leia (TGA);
    leia (TGO);
    NTOT ← NTOT + 1;
    se (TGO > 200) então
      NAC200 ← NAC200 + 1;
    fim-se
    se ( RDP > RDF ) então
      NRDP ← NRDP + 1;
    fim-se
    imprima ("Percentual (TGA+TGO)/(RDP+RDF) = ", 100*(TGA + TGO) / ( RDP + RDF ) );
    leia (RDP);
  fim-para
  PAC200 ← 100 * NAC200 / NTOT;
  imprima ("Percentual (TGO > 200 ) : ", PAC200);
  imprima ("Quantidade (RDP > RDF ) : ", NRDP);
fim
```



```
program l3p092;
var
  NAC200,NTOT,NRDP: integer;
  RDP,RDF,TGA,TGO,PAC200: real;
begin
  NAC200 := 0;
  NTOT := 0;
  NRDP := 0;
  write('Renda Pessoal: ');
  readLn(RDP);
  while ( RDP <> 0 ) do
  begin
    write('Renda Familiar: ');
    readLn(RDF);
    write('Total gasto com alimentacao: ');
    readLn(TGA);
    write('Total gasto com outras despesas: ');
    readLn(TGO);
    NTOT := NTOT + 1;
    if (TGO > 200) then
      NAC200 := NAC200 + 1;
    if ( RDP > RDF ) then
      NRDP := NRDP + 1;
    writeLn('Percentual (TGA+TGO)/(RDP+RDF) = ', 100*(TGA + TGO) / ( RDP + RDF ) );
    write('Renda Pessoal: ');
    read(RDP);
  end;
  PAC200 := 100 * NAC200 / NTOT;
  writeLn('Percentual (TGO > 200 ): ',PAC200:6:2);
  writeLn('Quantidade (RDP > RDF ): ',NRDP);
end.

% NAC200,NTOT,NRDP: integer;
% RDP,RDF,TGA,TGO,PAC200: real;
NAC200 = 0;
NTOT = 0;
NRDP = 0;
RDP = input('R Pessoal: ');
while ( RDP ~= 0 )
  RDF = input('R Familiar: ');
  TGA = input('Total gasto com alimentacao: ');
  TGO = input('Total gasto com outras despesas: ');
  NTOT = NTOT + 1;
  if (TGO > 200)
    NAC200 = NAC200 + 1;
  end
  if ( RDP > RDF )
    NRDP = NRDP + 1;
  end
  fprintf(1,'Percentual (TGA+TGO)/(RDP+RDF) = %.2f\n', 100*(TGA + TGO) / ( RDP + RDF ) );
  RDP = input('R Pessoal: ');
end
PAC200 = 100 * NAC200 / NTOT;
fprintf(1,'Percentual (TGO > 200 ): %.2f\n',PAC200);
fprintf(1,'Quantidade (RDP > RDF ): %d',NRDP);
```



93) Existem três candidatos a uma vaga no Senado. Feita a eleição, os votos são registrados em disco. O voto de cada eleitor foi codificado da seguinte forma:

1	Voto para os candidatos
2	
3	
0	Voto branco
4	Voto nulo

Deseja-se saber:

- O número do candidato vencedor;
- O número de votos em branco;
- O número de votos nulos;
- O número de eleitores que compareceram às urnas.

Admite-se que não são possíveis empates. Então, construa um algoritmo em PORTUGOL que execute esta tarefa, sabendo que o voto de cada eleitor é lido do disco através do comando **leia**(VOTO). O voto finalizador tem código -1, e não deve ser computado.

```
algoritmo L3P093;  
var  
  inteiro: VOTO, NTOT, NCAND1, NCAND2, NCAND3, NBRANCO, NNULO;  
início  
  NTOT ← 0;  
  NCAND1 ← 0;  
  NCAND2 ← 0;  
  NCAND3 ← 0;  
  NBRANCO ← 0;  
  NNULO ← 0;  
  leia(VOTO);  
  enquanto ( VOTO ≠ -1 ) faça  
    NTOT ← NTOT + 1;  
    se (VOTO = 1) então  
      NCAND1 ← NCAND1 + 1;  
    senão-se (VOTO = 2) então  
      NCAND2 ← NCAND2 + 1;  
    senão-se (VOTO = 3) então  
      NCAND3 ← NCAND3 + 1;  
    senão-se (VOTO = 0) então  
      NBRANCO ← NBRANCO + 1;  
    senão-se (VOTO = 4) então  
      NNULO ← NNULO + 1;  
    senão  
      NNULO ← NNULO + 1; { voto inválido = nulo}  
    fim-se  
  leia(VOTO);  
  fim-para  
  se ( NCAND1 > NCAND2 e NCAND1 > NCAND3) então  
    imprima("O candidato vencedor é o de número 1!");  
  senão-se ( NCAND2 > NCAND1 e NCAND2 > NCAND3) então  
    imprima("O candidato vencedor é o de número 2!");  
  senão-se ( NCAND3 > NCAND1 e NCAND3 > NCAND1) então  
    imprima("O candidato vencedor é o de número 3!");  
  senão  
    imprima("Houve empate entre candidatos! ***");  
  fim-se  
  imprima("Número de votos em branco: ",NBRANCO);  
  imprima("Número de votos nulos: ",NNULO);  
  imprima("Número de eleitores que compareceram às urnas: ",NTOT);  
fim
```



```
program l3p093;
var
  VOTO,NTOT,NCAND1,NCAND2,NCAND3,NBRANCO,NNULO: integer;
begin
  NTOT := 0;
  NCAND1 := 0;
  NCAND2 := 0;
  NCAND3 := 0;
  NBRANCO := 0;
  NNULO := 0;
  write('Voto: ');
  readLn(VOTO);
  while ( VOTO <> -1 ) do
  begin
    NTOT := NTOT + 1;
    if (VOTO = 1) then
      NCAND1 := NCAND1 + 1
    else if (VOTO = 2) then
      NCAND2 := NCAND2 + 1
    else if (VOTO = 3) then
      NCAND3 := NCAND3 + 1
    else if (VOTO = 0) then
      NBRANCO := NBRANCO + 1
    else if (VOTO = 4) then
      NNULO := NNULO + 1
    else
      NNULO := NNULO + 1; { voto inválido = nulo}
    write('Voto: ');
    readLn(VOTO);
  end;
  if ( NCAND1 > NCAND2 ) and ( NCAND1 > NCAND3 ) then
    writeLn('O candidato vencedor eh o de numero 1!')
  else if ( NCAND2 > NCAND1 ) and ( NCAND2 > NCAND3 ) then
    writeLn('O candidato vencedor eh o de numero 2!')
  else if ( NCAND3 > NCAND1 ) and ( NCAND3 > NCAND1 ) then
    writeLn('O candidato vencedor eh o de numero 3!')
  else
    writeLn('Houve empate entre candidatos! ***');
  writeLn('Numero de votos em branco: ',NBRANCO);
  writeLn('Numero de votos nulos: ',NNULO);
  writeLn('Numero de eleitores que compareceram às urnas: ',NTOT);
end.
```




```
%VOTO, NTOT, NCAND1, NCAND2, NCAND3, NBRANCO, NNULO: integer;
NTOT = 0;
NCAND1 = 0;
NCAND2 = 0;
NCAND3 = 0;
NBRANCO = 0;
NNULO = 0;
VOTO = input('Voto:');
while ( VOTO ~= -1 )
    NTOT = NTOT + 1;
    if (VOTO == 1)
        NCAND1 = NCAND1 + 1;
    elseif (VOTO == 2)
        NCAND2 = NCAND2 + 1;
    elseif (VOTO == 3)
        NCAND3 = NCAND3 + 1;
    elseif (VOTO == 0)
        NBRANCO = NBRANCO + 1;
    elseif (VOTO == 4)
        NNULO = NNULO + 1;
    else
        NNULO = NNULO + 1; % voto inválido = nulo
    end
    VOTO = input('Voto:');
end
if ( NCAND1 > NCAND2 ) & ( NCAND1 > NCAND3 )
    fprintf(1,'O candidato vencedor eh o de numero 1!\n');
elseif ( NCAND2 > NCAND1 ) & ( NCAND2 > NCAND3 )
    fprintf(1,'O candidato vencedor eh o de numero 2!\n');
elseif ( NCAND3 > NCAND1 ) & ( NCAND3 > NCAND1 )
    fprintf(1,'O candidato vencedor eh o de numero 3!\n');
else
    disp('Houve empate entre candidatos! ***');
end
fprintf(1,'Numero de votos em branco: %d\n',NBRANCO);
fprintf(1,'Numero de votos nulos: %d\n',NNULO);
fprintf(1,'Numero de eleitores que compareceram às urnas: %d\n',NTOT);
```



94) Escreva um algoritmo em PORTUGOL que calcule o imposto de renda de um grupo de contribuintes considerando:

- Os dados de cada contribuinte, número do CPF, número de dependentes e renda anual, serão digitados pelo usuário através do teclado;
- Para cada contribuinte será feito um desconto de R\$ 300,00 por dependente;
- Os valores da alíquota para cálculo do imposto são:

Renda Anual Líquida	Alíquota
Até R\$ 12.000,00	Isento
De R\$ 12.000,00 a R\$ 25.000,00	12%
Acima de R\$ 25.000,00	27,5%

- Para finalizar o algoritmo o CPF digitado zera igual a zero.

```
algoritmo L3P094;  
var  
  inteiro: CPF, NDEP;  
  real: RDANUAL, IMP;  
início  
  leia(CPF);  
  enquanto (CPF ≠ 0) faça  
    leia(NDEP);  
    leia(RDANUAL);  
    se (RDANUAL < 12000) então  
      IMP ← 0;  
    senão-se (RDANUAL < 25000) então  
      IMP ← RDANUAL * 12/100;  
    senão  
      IMP ← RDANUAL * 275/1000;  
    fim-se  
    IMP ← IMP - 300 * NDEP;  
    se (IMP >= 0) então  
      imprima("O Imposto de Renda a ser pago é: ", IMP);  
    senão  
      imprima("O Imposto de Renda a ser restituído é: ", -IMP);  
    fim-se  
  leia(CPF);  
fim-para  
fim
```



```
program l3p094;
var
  CPF, NDEP: integer;
  RDANUAL, IMP: real;
begin
  write('CPF: ');
  read(CPF);
  while (CPF <> 0 ) do
  begin
    write('Numero de dependentes: ');
    readLn(NDEP);
    write('Renda Anual: ');
    readLn(RDANUAL);
    if ( RDANUAL < 12000 ) then
      IMP := 0
    else if ( RDANUAL < 25000 ) then
      IMP := RDANUAL * 12/100
    else
      IMP := RDANUAL * 275/1000;
      IMP := IMP - 300 * NDEP;
      if ( IMP >= 0 ) then
        writeLn('O Imposto de Renda a ser pago eh: ',IMP)
      else
        writeLn('O Imposto de Renda a ser restituído eh: ',-IMP);
      write('CPF: ');
      readLn(CPF);
    end;
  end.

% CPF, NDEP: integer;
% RDANUAL, IMP: real;
CPF = input('CPF: ');
while (CPF ~= 0 )
  NDEP = input('Numero de dependentes: ');
  RDANUAL = input('Renda Anual: ');
  if ( RDANUAL < 12000 )
    IMP = 0;
  elseif ( RDANUAL < 25000 )
    IMP = RDANUAL * 12/100;
  else
    IMP = RDANUAL * 275/1000;
  end
  IMP = IMP - 300 * NDEP;
  if ( IMP >= 0 )
    fprintf(1,'O Imposto de R a ser pago eh: %.2f\n',IMP);
  else
    fprintf(1,'O Imposto de R a ser restituído eh: %.2f\n',-IMP);
  end
  CPF = input('CPF: ');
end
```



95) Em uma eleição presidencial, existem quatro candidatos. Os votos são informados através de código. Os dados utilizados para a escrutinagem obedecem à seguinte codificação:

- 1, 2, 3 e 4 = voto para os respectivos candidatos;
- 5 voto nulo;
- 6 voto em branco;

Elaborar um algoritmo em PORTUGOL que calcule e imprima:

- O total de votos para cada candidato;
- O total de votos nulos;
- O total de votos em branco;
- O percentual dos votos em branco e nulos sobre o total.

```
algoritmo L3P095;  
var  
  inteiro: VOTO, NTOT, NCAND1, NCAND2, NCAND3, NCAND4, NBRANCO, NNULO;  
  real: PERC;  
início  
  NTOT <- 0;  
  NCAND1 <- 0;  
  NCAND2 <- 0;  
  NCAND3 <- 0;  
  NCAND4 <- 0;  
  NBRANCO <- 0;  
  NNULO <- 0;  
  leia(VOTO);  
  enquanto ( VOTO >= 1 ) faça  
    NTOT <- NTOT + 1;  
    se (VOTO = 1) então  
      NCAND1 <- NCAND1 + 1;  
    senão-se (VOTO = 2) então  
      NCAND2 <- NCAND2 + 1;  
    senão-se (VOTO = 3) então  
      NCAND3 <- NCAND3 + 1;  
    senão-se (VOTO = 4) então  
      NCAND4 <- NCAND4 + 1;  
    senão-se (VOTO = 5) então  
      NNULO <- NNULO + 1;  
    senão-se (VOTO = 6) então  
      NBRANCO <- NBRANCO + 1;  
    senão  
      NNULO <- NNULO + 1; {voto inválido}  
    fim-se  
  leia(VOTO);  
fim-para  
  imprima("O total de votos para o candidato 1 é: ",NCAND1);  
  imprima("O total de votos para o candidato 2 é: ",NCAND2);  
  imprima("O total de votos para o candidato 3 é: ",NCAND3);  
  imprima("O total de votos para o candidato 4 é: ",NCAND4);  
  imprima("Número de votos em branco: ",NBRANCO);  
  imprima("Número de votos nulos: ",NNULO);  
  PERC <- (NBRANCO + NNULO) / NTOT;  
  imprima("Percentual de votos branco e nulos sobre o total é: ",PERC);  
fim
```



```
program l3p095;
var
  VOTO, NTOT, NCAND1, NCAND2, NCAND3, NCAND4, NBRANCO, NNULO: integer;
  PERC: real;
begin
  NTOT := 0;
  NCAND1 := 0;
  NCAND2 := 0;
  NCAND3 := 0;
  NCAND4 := 0;
  NBRANCO := 0;
  NNULO := 0;
  write('Voto: ');
  readLn(VOTO);
  while ( VOTO >= 1 ) do
  begin
    NTOT := NTOT + 1;
    if (VOTO = 1) then
      NCAND1 := NCAND1 + 1
    else if (VOTO = 2) then
      NCAND2 := NCAND2 + 1
    else if (VOTO = 3) then
      NCAND3 := NCAND3 + 1
    else if (VOTO = 4) then
      NCAND4 := NCAND4 + 1
    else if (VOTO = 5) then
      NNULO := NNULO + 1
    else if (VOTO = 6) then
      NBRANCO := NBRANCO + 1
    else
      NNULO := NNULO + 1; {voto inválido}
    write('Voto: ');
    readLn(VOTO);
  end;
  writeLn('O total de votos para o candidato 1 eh: ', NCAND1);
  writeLn('O total de votos para o candidato 2 eh: ', NCAND2);
  writeLn('O total de votos para o candidato 3 eh: ', NCAND3);
  writeLn('O total de votos para o candidato 4 eh: ', NCAND4);
  writeLn('Numero de votos em branco: ', NBRANCO);
  writeLn('Numero de votos nulos: ', NNULO);
  PERC := 100 * (NBRANCO + NNULO) / NTOT;
  writeLn('Percentual de votos branco e nulos sobre o total eh: ', PERC:6:2);
end.
```



```
NTOT = 0;
NCAND1 = 0;
NCAND2 = 0;
NCAND3 = 0;
NCAND4 = 0;
NBRANCO = 0;
NNULO = 0;
VOTO = input('Voto: ');
while ( VOTO >= 1 )
    NTOT = NTOT + 1;
    if (VOTO == 1)
        NCAND1 = NCAND1 + 1;
    elseif (VOTO == 2)
        NCAND2 = NCAND2 + 1;
    elseif (VOTO == 3)
        NCAND3 = NCAND3 + 1;
    elseif (VOTO == 4)
        NCAND4 = NCAND4 + 1;
    elseif (VOTO == 5)
        NNULO = NNULO + 1;
    elseif (VOTO == 6)
        NBRANCO = NBRANCO + 1;
    else
        NNULO = NNULO + 1; % voto inválido
    end
    VOTO = input('Voto: ');
end
fprintf(1,'O total de votos para o candidato 1 é: %d\n',NCAND1);
fprintf(1,'O total de votos para o candidato 2 é: %d\n',NCAND2);
fprintf(1,'O total de votos para o candidato 3 é: %d\n',NCAND3);
fprintf(1,'O total de votos para o candidato 4 é: %d\n',NCAND4);
fprintf(1,'Número de votos em branco: %d\n',NBRANCO);
fprintf(1,'Número de votos nulos: %d\n',NNULO);
PERC = 100 * (NBRANCO + NNULO) / NTOT;
fprintf(1,'Percentual de votos branco e nulos sobre o total é: %.2f\n',PERC);
```



96) Uma Empresa de fornecimento de energia elétrica faz a leitura mensal dos medidores de consumo. Para cada consumidor, são digitados os seguintes dados:

- Número do consumidor;
- Quantidade de kWh consumidos durante o mês;
- Tipo (código) do consumidor.
 - 1 – residencial, preço em reais por kWh = 0,3
 - 2 – comercial, preço em reais por kWh = 0,5
 - 3 – industrial, preço em reais por kWh = 0,7

Os dados devem ser lidos até que seja encontrado um consumidor com Número 0 (zero). Escreva um algoritmo em PORTUGOL que calcule e imprima:

- O custo total para cada consumidor;
- O total de consumo para os três tipos de consumidor;
- A média de consumo dos tipos 1 e 2.

```
algoritmo L3P096;
var
  inteiro: NUM, TIPO, NC1, NC2;
  real: QTDKWH, CUSTO, TOTAL1, TOTAL2, TOTAL3, MEDIA1, MEDIA2;
início
  NC1 ← 0;
  NC2 ← 0;
  TOTAL1 ← 0;
  TOTAL2 ← 0;
  TOTAL3 ← 0;
  leia (NUM);
  enquanto ( NUM ≠ 0 ) faça
    leia (QTDKWH);
    leia (TIPO);
    se ( TIPO = 1 ) então
      CUSTO ← QTDKWH * 0.3;
      NC1 ← NC1 + 1;
      TOTAL1 ← TOTAL1 + QTDKWH;
    senão-se ( TIPO = 2 ) então
      CUSTO ← QTDKWH * 0.5;
      NC2 ← NC2 + 1;
      TOTAL2 ← TOTAL2 + QTDKWH;
    senão
      CUSTO ← QTDKWH * 0.7;
      TOTAL3 ← TOTAL3 + QTDKWH;
  fim-se
  imprima ("O custo total desse consumidor é: ", CUSTO);
  leia (NUM);
fim-enquanto
imprima ("O total de consumo de consumidores do tipo 1 é: ", TOTAL1);
imprima ("O total de consumo de consumidores do tipo 2 é: ", TOTAL2);
imprima ("O total de consumo de consumidores do tipo 3 é: ", TOTAL3);
MEDIA1 ← TOTAL1 / NC1;
MEDIA2 ← TOTAL2 / NC2;
imprima ("A média de consumo do consumo de tipo 1 é: ", MEDIA1);
imprima ("A média de consumo do consumo de tipo 2 é: ", MEDIA2);
fim
```



```
program l3p096;
var
  NUM, TIPO, NC1, NC2: integer;
  QTDKWH, CUSTO, TOTAL1, TOTAL2, TOTAL3, MEDIA1, MEDIA2: real;
begin
  NC1 := 0;
  NC2 := 0;
  TOTAL1 := 0;
  TOTAL2 := 0;
  TOTAL3 := 0;
  write('Numero do consumidor: ');
  readLn(NUM);
  while ( NUM <> 0 ) do
  begin
    write('Consumo (kWh): ');
    readLn(QTDKWH);
    write('Tipo de consumidor (1-Residencial/2-Comercial/3-Industrial): ');
    read(TIPO);
    if ( TIPO = 1 ) then begin
      CUSTO := QTDKWH * 0.3;
      NC1 := NC1 + 1;
      TOTAL1 := TOTAL1 + QTDKWH; end
    else if ( TIPO = 2 ) then begin
      CUSTO := QTDKWH * 0.5;
      NC2 := NC2 + 1;
      TOTAL2 := TOTAL2 + QTDKWH; end
    else begin
      CUSTO := QTDKWH * 0.7;
      TOTAL3 := TOTAL3 + QTDKWH; end;
    writeLn('O custo total desse consumidor eh: ', CUSTO:3:2);
    write('Numero do consumidor: ');
    readLn(NUM);
  end;
  writeLn('O total de consumo de consumidores do tipo 1 eh: ', TOTAL1:3:2);
  writeLn('O total de consumo de consumidores do tipo 2 eh: ', TOTAL2:3:2);
  writeLn('O total de consumo de consumidores do tipo 3 eh: ', TOTAL3:3:2);
  MEDIA1 := TOTAL1 / NC1;
  MEDIA2 := TOTAL2 / NC2;
  writeLn('A media de consumo do consumidor tipo 1 eh: ', MEDIA1:3:2);
  writeLn('A media de consumo do consumidor tipo 2 eh: ', MEDIA2:3:2);
end.
```




```
% NUM, TIPO, NC1, NC2: integer;
% QTDKWH, CUSTO, TOTAL1, TOTAL2, TOTAL3, MEDIA1, MEDIA2: real;
NC1 = 0;
NC2 = 0;
TOTAL1 = 0;
TOTAL2 = 0;
TOTAL3 = 0;
NUM = input('Numero do consumidor: ');
while ( NUM ~= 0 )
    QTDKWH = input('Consumo (kWh): ');
    disp('Tipo de consumidor (1-Residencial/2-Comercial/3-Industrial): ');
    TIPO = input('Escolha um: ');
    if ( TIPO == 1 )
        CUSTO = QTDKWH * 0.3;
        NC1 = NC1 + 1;
        TOTAL1 = TOTAL1 + QTDKWH;
    elseif ( TIPO == 2 )
        CUSTO = QTDKWH * 0.5;
        NC2 = NC2 + 1;
        TOTAL2 = TOTAL2 + QTDKWH;
    else
        CUSTO = QTDKWH * 0.7;
        TOTAL3 = TOTAL3 + QTDKWH;
    end
    end
    fprintf(1, 'O custo total desse consumidor eh: %.2f\n', CUSTO);
    NUM = input('Numero do consumidor: ');
end;
fprintf(1, 'O total de consumo de consumidores do tipo 1 eh: %.2f\n', TOTAL1);
fprintf(1, 'O total de consumo de consumidores do tipo 2 eh: %.2f\n', TOTAL2);
fprintf(1, 'O total de consumo de consumidores do tipo 3 eh: %.2f\n', TOTAL3);
MEDIA1 = TOTAL1 / NC1;
MEDIA2 = TOTAL2 / NC2;
fprintf(1, 'A media de consumo do consumidor tipo 1 eh: %.2f\n', MEDIA1);
fprintf(1, 'A media de consumo do consumidor tipo 2 eh: %.2f\n', MEDIA2);
```



97) Escreva um algoritmo em PORTUGOL que receba o número da conta e o saldo de várias pessoas. O algoritmo deve imprimir todas as contas, os respectivos saldos e uma das mensagens: positivo/negativo. Ao final, o percentual de pessoas com saldo negativo. O algoritmo acaba quando se digita um número negativo para a conta.

```
algoritmo L3P097;  
var  
  inteiro: NCONTA, NTOT, NNEG;  
  real: SALDO, PERC;  
início  
  NTOT ← 0;  
  NNEG ← 0;  
  leia(NCONTA);  
  enquanto ( NCONTA >= 0 ) faça  
    leia(SALDO);  
    NTOT ← NTOT + 1;  
    imprima(NCONTA);  
    imprima(SALDO);  
    se ( SALDO > 0 ) então  
      imprima( "POSITIVO" );  
    senão-se ( SALDO < 0 ) então  
      imprima( "NEGATIVO" );  
      NNEG ← NNEG + 1;  
    fim-se  
  leia(NCONTA);  
fim-enquanto  
  PERC ← NNEG / NTOT;  
  imprima( "O percentual de pessoas com saldo negativo: ", PERC );  
fim
```

```
program l3p097;  
var  
  NCONTA, NTOT, NNEG: integer;  
  SALDO, PERC: real;  
begin  
  NTOT := 0;  
  NNEG := 0;  
  write('Numero da conta: ');  
  readLn(NCONTA);  
  while ( NCONTA >= 0 ) do  
    begin  
      write('Saldo: ');  
      readLn(SALDO);  
      NTOT := NTOT + 1;  
      write('Conta: ', NCONTA, ', Saldo: ', SALDO:3:2);  
      if ( SALDO > 0 ) then  
        writeLn(' POSITIVO')  
      else if ( SALDO < 0 ) then begin  
        writeLn(' NEGATIVO');  
        NNEG := NNEG + 1; end;  
      write('Numero da conta: ');  
      readLn(NCONTA);  
    end;  
  PERC := 100 * NNEG / NTOT;  
  writeLn('O percentual de pessoas com saldo negativo: ', PERC:6:2);  
end.
```



```
% NCONTA,NTOT,NNEG: integer;
% SALDO,PERC: real;
NTOT = 0;
NNEG = 0;
NCONTA = input('Numero da conta: ');
while ( NCONTA >= 0 )
    SALDO = input('Saldo: ');
    NTOT = NTOT + 1;
    fprintf('Conta: %d, Saldo: %.2f\n',NCONTA,SALDO);
    if ( SALDO > 0 )
        disp(' POSITIVO');
    elseif ( SALDO < 0 )
        disp(' NEGATIVO')
        NNEG = NNEG + 1;
    end
    NCONTA = input('Numero da conta: ');
end
PERC = 100 * NNEG / NTOT;
fprintf(1,'O percentual de pessoas com saldo negativo: %.2f\n',PERC);
```



- 98) Uma agência de uma cidade do interior tem, no máximo, 10000 clientes. Criar um algoritmo em PORTUGOL que possa receber o número da conta, nome e saldo de cada cliente. Esse algoritmo deve imprimir todas as contas, os respectivos saldos e uma das mensagens: positivo/negativo. A digitação acaba quando se digita -999 para um número da conta ou quando chegar a 10000. Ao final, deverá sair o total de clientes com saldo negativo, o total de clientes da agência e o saldo da agência.

```
algoritmo L3P098;  
var  
  inteiro: NCONTA, NTOT, NNEG;  
  real: SALDO, SALDOAG;  
  literal: NOME;  
início  
  NTOT ← 0;  
  NNEG ← 0;  
  SALDOAG ← 0;  
  leia(NCONTA);  
  enquanto ( NTOT < 10000 e NCONTA ≠ -999 ) faça  
    leia(NOME);  
    leia(SALDO);  
    imprima(NCONTA);  
    imprima(SALDO);  
    se ( SALDO > 0 ) então  
      imprima("POSITIVO");  
    senão-se ( SALDO < 0 ) então  
      imprima("NEGATIVO");  
      NNEG ← NNEG + 1;  
    fim-se  
    NTOT ← NTOT + 1;  
    SALDOAG ← SALDOAG + SALDO;  
    leia(NCONTA);  
  fim-enquanto  
  imprima("O total de clientes com saldo negativo é: ",NNEG);  
  imprima("O total de clientes da agência é: ",NTOT);  
  imprima("O saldo da agência é: ",SALDOAG);  
fim
```



```
program l3p098;
var
  NCONTA,NTOT,NNEG: integer;
  SALDO,SALDOAG: real;
  NOME: string;
begin
  NTOT := 0;
  NNEG := 0;
  SALDOAG := 0;
  write('Numero da conta: ');
  readLn(NCONTA);
  while ( NTOT < 10000 ) and ( NCONTA <> -999 ) do
  begin
    write('Nome: ');
    readLn(NOME);
    write('Saldo: ');
    readLn(SALDO);
    write('Conta: ',NCONTA,', Saldo: ',SALDO:3:2);
    if ( SALDO > 0 ) then
      writeLn(' POSITIVO')
    else if ( SALDO < 0 ) then begin
      writeLn(' NEGATIVO');
      NNEG := NNEG + 1; end;
    NTOT := NTOT + 1;
    SALDOAG := SALDOAG + SALDO;
    write('Numero da conta: ');
    readLn(NCONTA);
  end;
  writeLn('O total de clientes com saldo negativo eh: ',NNEG);
  writeLn('O total de clientes da agencia eh: ',NTOT);
  writeLn('O saldo da agencia eh: ',SALDOAG:3:2);
end.

% NCONTA,NTOT,NNEG: integer;
% SALDO,SALDOAG: real;
% NOME: string;
NTOT = 0;
NNEG = 0;
SALDOAG = 0;
NCONTA = input('Numero da conta: ');
while ( NTOT < 10000 ) & ( NCONTA ~= -999 )
  NOME = input('Nome: ','s');
  SALDO = input('Saldo: ');
  fprintf('Conta: %d Saldo: %.2f\n',NCONTA,SALDO);
  if ( SALDO > 0 )
    disp(' POSITIVO')
  elseif ( SALDO < 0 )
    disp(' NEGATIVO');
    NNEG = NNEG + 1;
  end
  NTOT = NTOT + 1;
  SALDOAG = SALDOAG + SALDO;
  NCONTA = input('Numero da conta: ');
end;
fprintf(1,'O total de clientes com saldo negativo eh: %d\n',NNEG);
fprintf(1,'O total de clientes da agencia eh: %d\n',NTOT);
fprintf(1,'O saldo da agencia eh: %.2f\n',SALDOAG);
```



99) Criar um algoritmo em PORTUGOL que possa ler um conjunto de pedidos de compra e calcule o valor total da compra. Cada pedido é composto pelos seguintes campos:

- Número de pedido;
- Data do pedido (dia, mês, ano);
- Preço unitário;
- Quantidade.

O algoritmo deverá processar novos pedidos até que o usuário digite (zero) como número de pedido.

```
algoritmo L3P099;  
var  
  inteiro: NPED, DIA, MÊS, ANO, QTD;  
  real:   PRECOUNIT, PRECOTOTAL, VALORTOTAL;  
início  
  VALORTOTAL ← 0;  
  leia(NPED);  
  enquanto ( NPED ≠ 0 ) faça  
    leia(DIA, MÊS, ANO);  
    leia(PRECOUNIT);  
    leia(QTD);  
    PRECOTOT ← PRECOUNIT * QTD;  
    VALORTOTAL ← VALORTOTAL + PRECOTOT;  
    imprima("O preço total desse pedido é: ", PRECOTOT);  
    leia(NPED);  
  fim-enquanto  
  imprima("O total geral dos pedidos é: ", VALORTOTAL);  
fim
```

```
program l3p099;  
var  
  NPED, DIA, MES, ANO, QTD: integer;  
  PRECOUNIT, PRECOTOTAL, VALORTOTAL: real;  
begin  
  VALORTOTAL := 0;  
  write('Numero do pedido: ');  
  readLn(NPED);  
  while ( NPED <> 0 ) do  
    begin  
      write('Data: (Dia Mes Ano): ');  
      readLn(DIA, MES, ANO);  
      write('Preco unitario: ');  
      readLn(PRECOUNIT);  
      write('Quantidade: ');  
      readLn(QTD);  
      PRECOTOTAL := PRECOUNIT * QTD;  
      VALORTOTAL := VALORTOTAL + PRECOTOTAL;  
      writeLn('O preco total desse pedido eh: ', PRECOTOTAL:3:2);  
      write('Numero do pedido: ');  
      readLn(NPED);  
    end;  
  writeLn('O total geral dos pedidos eh: ', VALORTOTAL:3:2);  
end.
```



```
% NPED, DIA, MES, ANO, QTD: integer;
% PRECUNIT,PRECOTOTAL,VALORTOTAL: real;
VALORTOTAL = 0;
NPED = input('Numero do pedido: ');
while ( NPED ~= 0 )
    DIA = input('Dia: ');
    MES = input('Mes: ');
    ANO = input('Ano: ');
    PRECUNIT = input('Preco unitario: ');
    QTD = input('Quantidade: ');
    PRECOTOTAL = PRECUNIT * QTD;
    VALORTOTAL = VALORTOTAL + PRECOTOTAL;
    fprintf(1,'O preco total desse pedido eh: %.2f\n',PRECOTOTAL);
    NPED = input('Numero do pedido: ');
end
fprintf(1,'O total geral dos pedidos eh: %.2f\n',VALORTOTAL);
```



100) Uma fábrica produz e vende vários produtos e para cada um deles tem-se o nome, quantidade produzida e quantidade vendida. Criar um algoritmo em PORTUGOL que imprima:

- Para cada produto, nome, quantidade no estoque e uma mensagem se o produto tiver menos de 50 itens no estoque;
- Nome e quantidade do produto com maior estoque;

```
algoritmo L3P100;  
var  
  inteiro: QTDPROD, QTDVEND, ESTOQUE, ESTOQUEMAIOR;  
  literal: NOME, NOMEMAIOR;  
inicio  
  ESTOQUEMAIOR ← 0;  
  leia (NOME);  
  enquanto ( NOME ≠ "@" ) faça  
    leia (QTDPROD, QTDVEND);  
    ESTOQUE ← QTDPROD - QTDVEND;  
    imprima ("Nome: ", NOME);  
    imprima ("Estoque: ", ESTOQUE);  
    se ( ESTOQUE < 50 ) então  
      imprima ("Estoque em baixa");  
    fim-se  
    se ( ESTOQUE > ESTOQUEMAIOR ) então  
      ESTOQUEMAIOR ← ESTOQUE;  
      NOMEMAIOR ← NOME;  
    fim-se  
  leia (NOME);  
fim-enquanto  
imprima ("Produto: ", NOMEMAIOR, " com maior estoque: ", ESTOQUEMAIOR);  
fim
```

```
program l3p100;  
var  
  QTDPROD, QTDVEND, ESTOQUE, ESTOQUEMAIOR: integer;  
  NOME, NOMEMAIOR: string;  
begin  
  ESTOQUEMAIOR := 0;  
  write('Nome: ');  
  readLn(NOME);  
  while ( NOME <> '@' ) do  
    begin  
      write('Quantidade Produzida: ');  
      readLn(QTDPROD);  
      write('Quantidade Vendida: ');  
      readLn(QTDVEND);  
      ESTOQUE := QTDPROD - QTDVEND;  
      writeLn('Nome: ', NOME);  
      writeLn('Estoque: ', ESTOQUE);  
      if ( ESTOQUE < 50 ) then  
        writeLn('Estoque em baixa');  
      if ( ESTOQUE > ESTOQUEMAIOR ) then begin  
        ESTOQUEMAIOR := ESTOQUE;  
        NOMEMAIOR := NOME; end;  
      write('Nome: ');  
      readLn(NOME);  
    end;  
  writeLn('Produto: ', NOMEMAIOR, ' com maior estoque: ', ESTOQUEMAIOR);  
end.
```




```
ESTOQUEMAIOR = 0;
NOME = input('Nome: ','s');
while ( NOME ~= '@' )
    QTDPROD = input('Quantidade Produzida: ');
    QTDVEND = input('Quantidade Vendida: ');
    ESTOQUE = QTDPROD - QTDVEND;
    fprintf(1,'Nome: %s\n',NOME);
    fprintf(1,'Estoque: %d\n',ESTOQUE);
    if ( ESTOQUE < 50 )
        disp('Estoque em baixa');
    end
    if (ESTOQUE > ESTOQUEMAIOR )
        ESTOQUEMAIOR = ESTOQUE;
        NOMEMAIOR = NOME;
    end
    NOME = input('Nome: ','s');
end
fprintf(1,'Produto: %s com maior estoque: %d\n',NOMEMAIOR,ESTOQUEMAIOR);
```