

No exemplo anterior, a variável **a** é inicializada com o valor 15 e vai sendo decrementada em duas unidades e, a cada decremento, o bloco de comando que está entre as chaves { ... } é executado. Esse processo se repete até o valor da variável **a** se tornar menor que 1 (quando a condição $a \geq 1$ assumir valor falso).

4.3.2 ESTRUTURA DE REPETIÇÃO WHILE

Trata-se de uma estrutura de repetição que pode ser utilizada quando o número de repetições necessárias não é fixo. Os comandos serão repetidos até a condição assumir o valor falso.

Nesse tipo de estrutura o teste condicional ocorre no início, isso significa que existe a possibilidade da repetição não ser executada, quando a condição assumir valor falso logo na primeira verificação.

```
while (condição)
    comando;
```

Enquanto a **condição** for verdadeira o **comando** será executado.

```
while (condição)
{
    comando1;
    comando2;
    comando3;
    ...
}
```

Enquanto a **condição** for verdadeira, os **comandos** que estão dentro das chaves serão executados (**comando1**, **comando2**, **comando3**, ...).

4.3.3 ESTRUTURA DE REPETIÇÃO DO-WHILE

Trata-se de uma estrutura de repetição que pode ser utilizada quando o número de repetições necessárias não é fixo. Os comandos serão repetidos até a condição assumir o valor falso.

Nesse tipo de estrutura o teste condicional ocorre no fim. Isso significa que a repetição será executada no mínimo uma vez, quando todo o bloco for executado uma vez e, ao final, a condição assumir valor falso.

```
do
{
    comandos;
}
while (condição);
```

Os **comandos** serão repetidos até que a **condição** assuma valor falso.

EXERCÍCIOS RESOLVIDOS

1. Um funcionário de uma empresa recebe aumento salarial anualmente. Sabe-se que:
 - a) esse funcionário foi contratado em 1995, com salário inicial de R\$ 1.000,00;
 - b) em 1996 recebeu aumento de 1,5% sobre seu salário inicial;
 - c) a partir de 1997 (inclusive), os aumentos salariais sempre corresponderam ao dobro do percentual do ano anterior.

Faça um programa que determine o salário atual desse funcionário.

ALGORITMO**Solução:**

```

ALGORITMO
DECLARE i, ano_atual, salario, novo_salario, percentual NUMÉRICO
LEIA ano_atual
salario ← 1000
percentual ← 1,5%
novo_salario ← salario + percentual * salario
PARA i ← 1997 ATÉ ano_atual FAÇA
    INÍCIO
        percentual ← 2 * percentual
        novo_salario ← novo_salario + percentual * novo_salario
    FIM
ESCREVA novo_salario
FIM_ALGORITMO.

```

**RESOLUÇÃO
PASCAL****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX1_A.PAS e \EXERC\CAP4\PASCAL\EX1_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX1_B.PAS e \EXERC\CAP4\PASCAL\EX1_B.EXE

**RESOLUÇÃO
C/C++****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX1_A.CPP e \EXERC\CAP4\C++\EX1_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX1_B.CPP e \EXERC\CAP4\C++\EX1_B.EXE

2. Faça um programa que leia um valor N inteiro e positivo, calcule e mostre o valor de E, conforme a fórmula a seguir:

$$E = 1 + 1/1! + 1/2! + 1/3! + \dots + 1/N!$$

ALGORITMO**Solução:**

```

ALGORITMO
DECLARE n, e, i, j, fat NUMÉRICO
LEIA n
e ← 1
PARA i ← 1 ATÉ n FAÇA
    INÍCIO
        fat ← 1
        PARA j ← 1 ATÉ i FAÇA
            INÍCIO
                fat ← fat * j
            FIM
        e ← e + 1/fat
    FIM
ESCREVA e
FIM_ALGORITMO.

```

**RESOLUÇÃO
PASCAL****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX2_A.PAS e \EXERC\CAP4\PASCAL\EX2_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX2_B.PAS e \EXERC\CAP4\PASCAL\EX2_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX2_A.CPP e \EXERC\CAP4\C++\EX2_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX2_B.CPP e \EXERC\CAP4\C++\EX2_B.EXE

3. Faça um programa que leia um número N e que indique quantos valores inteiros e positivos devem ser lidos a seguir. Para cada número lido, mostre uma tabela contendo o valor lido e o fatorial desse valor.

**ALGORITMO****SOLUÇÃO:**

```
ALGORITMO
DECLARE n, num, i, j, fat NUMÉRICO
LEIA n
PARA i ← 1 ATÉ n FAÇA
  INÍCIO
    LEIA num
    fat ← 1
    PARA j ← 1 ATÉ num FAÇA
      INÍCIO
        fat ← fat * j
      FIM
    ESCRIVA fat
  FIM
FIM_ALGORITMO.
```

**RESOLUÇÃO
PASCAL****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX3_A.PAS e \EXERC\CAP4\PASCAL\EX3_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX3_B.PAS e \EXERC\CAP4\PASCAL\EX3_B.EXE

**RESOLUÇÃO
C/C++****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX3_A.CPP e \EXERC\CAP4\C++\EX3_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX3_B.CPP e \EXERC\CAP4\C++\EX3_B.EXE

4. Faça um programa que leia cinco pares de valores (a,b), todos inteiros e positivos, um de cada vez. Mostre os números inteiros pares de a até b (inclusive).

**ALGORITMO****SOLUÇÃO:**

```
ALGORITMO
DECLARE cont, a, b, i NUMÉRICO
PARA CONT ← 1 ATÉ 5 FAÇA
```

```

INÍCIO
LEIA a, b
PARA i ← a ATÉ b FAÇA
    INÍCIO
    SE RESTO(i/2) = 0
    ENTÃO ESCREVA i
    FIM
FIM
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX4_A.PAS e \EXERC\CAP4\PASCAL\EX4_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX4_B.PAS e \EXERC\CAP4\PASCAL\EX4_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX4_A.CPP e \EXERC\CAP4\C++\EX4_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX4_B.CPP e \EXERC\CAP4\C++\EX4_B.EXE

5. Faça um programa que leia dez conjuntos de dois valores, o primeiro representando o número do aluno e o segundo representando a sua altura em centímetros. Encontre o aluno mais alto e o mais baixo. Mostre o número do aluno mais alto e o número do mais baixo, junto com suas alturas.



ALGORITMO

Solução:

```

ALGORITMO
DECLARE cont, num, alt, maior, num_maior, menor, num_menor NUMÉRICO
PARA cont ← 1 ATÉ 10 FAÇA
    INÍCIO
    LEIA num, alt
    SE cont = 1
    ENTÃO INÍCIO
        maior ← alt
        num_maior ← num
        menor ← alt
        num_menor ← num
    FIM
    SENÃO INÍCIO
        SE alt > maior
        ENTÃO INÍCIO
            maior ← alt
            num_maior ← num
        FIM
        SE alt < menor
        ENTÃO INÍCIO
            menor ← alt
            num_menor ← num
        FIM
    FIM
FIM
ESCREVA maior, num_maior

```

```

    ESCRIVA menor, num_menor
    FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

```

\EXERC\CAP4\PASCAL\EX5_A.PAS e \EXERC\CAP4\PASCAL\EX5_A.EXE

```

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

```

\EXERC\CAP4\PASCAL\EX5_B.PAS e \EXERC\CAP4\PASCAL\EX5_B.EXE

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

```

\EXERC\CAP4\C++\EX5_A.CPP e \EXERC\CAP4\C++\EX5_A.EXE

```

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

```

\EXERC\CAP4\C++\EX5_B.CPP e \EXERC\CAP4\C++\EX5_B.EXE

```

6. Foi feita uma estatística em cinco cidades brasileiras para coletar dados sobre acidentes de trânsito. Foram obtidos os seguintes dados:

- código da cidade;
- número de veículos de passeio (em 1999);
- número de acidentes de trânsito com vítimas (em 1999).

Deseja-se saber:

- qual o maior e o menor índice de acidentes de trânsito e a que cidades pertencem;
- qual a média de veículos nas cinco cidades juntas;
- qual a média de acidentes de trânsito nas cidades com menos de 2.000 veículos de passeio.



SOLUÇÃO:

```

ALGORITMO
DECLARE cont, cod, num_vei, num_acid NUMÉRICO
        maior, cid_maior, menor, cid_menor NUMÉRICO
        media_vei, soma_vei, media_acid NUMÉRICO
        soma_acid, cont_acid NUMÉRICO

soma_vei ← 0
soma_acid ← 0
cont_acid ← 0
PARA cont ← 1 ATÉ 5 FAÇA
    INÍCIO
        LEIA cod, num_vei, num_acid
        SE cont = 1
            ENTÃO INÍCIO
                maior ← num_acid
                cid_maior ← cod
                menor ← num_acid
                cid_menor ← cod
            FIM
        SENÃO INÍCIO
            SE num_acid > maior
                ENTÃO INÍCIO
                    maior ← num_acid
                    cid_maior ← cod
                FIM
        FIM
    FIM
FIM

```

```

SE num_acid < menor
ENTÃO INÍCIO
    menor ← num_acid
    cid_menor ← cod
FIM
FIM
soma_vei ← soma_vei + num_vei
SE num_vei < 2000
ENTÃO INÍCIO
    soma_acid ← soma_acid + num_acid
    cont_acid ← cont_acid + 1
FIM
FIM
ESCREVA maior, cid_maior
ESCREVA menor, cid_menor
media_vei ← soma_vei/5
ESCREVA media_vei
SE cont_acid = 0
    ENTÃO ESCREVA "Não foi digitada nenhuma cidade com menos de 2000
    veículos"
    SENÃO INÍCIO
        media_acid ← soma_acid/cont_acid
        ESCREVA media_acid
    FIM
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX6_A.PAS e \EXERC\CAP4\PASCAL\EX6_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX6_B.PAS e \EXERC\CAP4\PASCAL\EX6_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX6_A.CPP e \EXERC\CAP4\C++\EX6_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX6_B.CPP e \EXERC\CAP4\C++\EX6_B.EXE

7. Faça um programa que leia o número de termos e um valor positivo para X, calcule e mostre o valor da série a seguir:

$$S = -\frac{X^2}{1!} + \frac{X^3}{2!} - \frac{X^4}{3!} + \frac{X^5}{4!} - \frac{X^6}{3!} + \frac{X^7}{2!} - \frac{X^8}{1!} + \frac{X^9}{2!} - \frac{X^{10}}{3!} + \frac{X^{11}}{4!} - \dots$$



SOLUÇÃO:

```

ALGORITMO
DECLARE fim, i, j, x, expoente, num_termos NUMÉRICO
den, denominador, fat, s NUMÉRICO
LEIA num_termos, x
s ← 0
denominador ← 1
PARA i ← 1 TO num_termos FAÇA
    INÍCIO

```

```

fim ← denominador
fat ← 1
PARA j ← 1 ATÉ fim FAÇA
    INÍCIO
        fat ← fat * j
    FIM
expoente ← i + 1
SE RESTO (expoente / 2) = 0
    ENTÃO s ← s - xexpoente/fat
    SENÃO s ← s + xexpoente / fat
SE denominador = 4
    ENTÃO den ← -1
SE denominador = 1
    ENTÃO den ← 1
SE den = 1
    ENTÃO denominador ← denominador + 1
    SENÃO denominador ← denominador - 1
FIM
ESCREVA s
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX7_A.PAS e \EXERC\CAP4\PASCAL\EX7_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX7_B.PAS e \EXERC\CAP4\PASCAL\EX7_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX7_A.CPP e \EXERC\CAP4\C++\EX7_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX7_B.CPP e \EXERC\CAP4\C++\EX7_B.EXE

8. Uma empresa possui dez funcionários com as seguintes características: código, número de horas trabalhadas no mês, turno de trabalho (M – Matutino, V – Vespertino ou N – Noturno), categoria (O – Operário ou G – Gerente), valor da hora trabalhada. Sabendo-se que essa empresa deseja informatizar sua folha de pagamento, faça um programa que:

- leia as informações dos funcionários, exceto o valor da hora trabalhada, não permitindo que sejam informados turnos nem categorias inexistentes. Trabalhar sempre com a digitação de letras maiúsculas;
- calcule o valor da hora trabalhada, conforme a tabela a seguir.

| CATEGORIA | TURNO | VALOR DA HORA TRABALHADA |
|-----------|--------|--------------------------|
| G | N | 18% do salário mínimo |
| G | M ou V | 15% do salário mínimo |
| O | N | 13% do salário mínimo |
| O | M ou V | 10% do salário mínimo |

Adote o valor de R\$ 150,00 para o salário mínimo.

- calcule o salário inicial dos funcionários com base no valor da hora trabalhada e no número de horas trabalhadas;

- d) calcule o valor do auxílio-alimentação recebido por funcionário de acordo com o seu salário inicial, conforme a tabela a seguir.

| SALÁRIO INICIAL | AUXÍLIO-ALIMENTAÇÃO |
|-------------------------------|------------------------|
| Até R\$ 300,00 | 20% do salário inicial |
| Entre R\$ 300,00 e R\$ 600,00 | 15% do salário inicial |
| Acima de R\$ 600,00 | 5% do salário inicial |

- e) mostre o código, número de horas trabalhadas, valor da hora trabalhada, salário inicial, auxílio-alimentação e o salário final (salário inicial + auxílio-alimentação).

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
DECLARE cont, codigo, nht, valor NUMÉRICO
sal_min, sal_inicial, aux, sal_final NUMÉRICO
turno, categoria LITERAL
sal_min ← 150
PARA cont ← 1 ATÉ 10 FAÇA
  INÍCIO
    LEIA codigo, nht, turno, categoria
    ENQUANTO (turno ≠ "M") E (turno ≠ "V") E (turno ≠ "N") FAÇA
      INÍCIO
        LEIA turno
      FIM
    ENQUANTO (categoria ≠ "G") E (categoria ≠ "O") FAÇA
      INÍCIO
        LEIA categoria
      FIM
    SE categoria = "G"
      ENTÃO INÍCIO
        SE turno = "N"
          ENTÃO valor ← 18% * sal_min
          SENÃO valor ← 15% * sal_min
        FIM
      SENÃO INÍCIO
        SE turno = "N"
          ENTÃO valor ← 13% * sal_min
          SENÃO valor ← 10% * sal_min
        FIM
    sal_inicial ← nht * valor
    SE sal_inicial ≤ 300
      ENTÃO aux ← 20% * sal_inicial
    SENÃO SE sal_inicial < 600
      ENTÃO aux ← 15% * sal_inicial
      SENÃO aux ← 5% * sal_inicial
    sal_final ← sal_inicial + aux
    ESCREVA codigo, nht, valor, sal_inicial, aux, sal_final
  FIM
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX8_A.PAS e \EXERC\CAP4\PASCAL\EX8_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX8_B.PAS e \EXERC\CAP4\PASCAL\EX8_B.EXE

RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX8_A.CPP e \EXERC\CAP4\C++\EX8_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX8_B.CPP e \EXERC\CAP4\C++\EX8_B.EXE

9. Uma empresa contratou 15 funcionários temporários. De acordo com o valor das vendas mensais, os funcionários adquirem pontos que determinarão seus salários ao final de cada mês. Sabe-se que esses funcionários trabalharão nos meses de novembro a janeiro do ano subsequente.

Faça um programa que:

- leia as pontuações nos três meses de cada funcionário;
- calcule e mostre a pontuação geral de cada funcionário nos três meses;
- calcule e mostre a média das pontuações de cada funcionário nos três meses;
- determine e mostre a maior pontuação atingida entre todos os funcionários nos três meses.



ALGORITMO

SOLUÇÃO:

```

ALGORITMO
DECLARE cont_func, cont_mes, pont NUMÉRICO
pont_total, maior_pont, media_pont NUMÉRICO
maior_pont ← 0
PARA cont_func ← 1 ATÉ 15 FAÇA
  INÍCIO
    pont_total ← 0
    PARA cont_mes ← 1 ATÉ 3 FAÇA
      INÍCIO
        LEIA pont
        pont_total ← pont_total + pont
        SE pont > maior_pont
          ENTÃO maior_pont ← pont
      FIM
    FIM
    ESCREVA pont_total
    media_pont ← pont_total/3
    ESCREVA media_pont
  FIM
ESCREVA maior_pont
FIM_ALGORITMO.

```

RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX9_A.PAS e \EXERC\CAP4\PASCAL\EX9_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX9_B.PAS e \EXERC\CAP4\PASCAL\EX9_B.EXE

RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX9_A.CPP e \EXERC\CAP4\C++\EX9_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX9_B.CPP e \EXERC\CAP4\C++\EX9_B.EXE

10. Faça um programa que monte os oito primeiros termos da sequência de Fibonacci.

0-1-1-2-3-5-8-13-21-34-55- ...

ALGORITMO**SOLUÇÃO:**

```
ALGORITMO
    DECLARE cont, num1, num2, res NUMÉRICO
    num1 ← 0
    num2 ← 1
    ESCRIVA num1
    ESCRIVA num2
    PARA cont ← 3 ATÉ 8 FAÇA
        INÍCIO
            res ← num1 + num2
            ESCRIVA res
            num1 ← num2
            num2 ← res
        FIM
    FIM_ALGORITMO.
```

**RESOLUÇÃO
PASCAL****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX10_A.PAS e \EXERC\CAP4\PASCAL\EX10_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX10_B.PAS e \EXERC\CAP4\PASCAL\EX10_B.EXE

**RESOLUÇÃO
C/C++****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX10_A.CPP e \EXERC\CAP4\C++\EX10_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX10_B.CPP e \EXERC\CAP4\C++\EX10_B.EXE

11. Faça um programa que leia o número de termos, determine e mostre os valores de acordo com a série abaixo:

Série = 2, 7, 3, 4, 21, 12, 8, 63, 48, 16, 189, 192, 32, 567, 768, 64, ...

ALGORITMO**SOLUÇÃO:**

```
ALGORITMO
    DECLARE i, num_termos, num1, num2, num3 NUMÉRICO
    LEIA num_termos
    num1 ← 2
    num2 ← 7
    num3 ← 3
    ESCRIVA num1
    ESCRIVA num2
    ESCRIVA num3
    i ← 4
    enquanto i ≠ num_termos FAÇA
```

```

INÍCIO
num1 ← num1 * 2
ESCREVA num1
i ← i + 1
SE i ≠ num_termos
ENTÃO INÍCIO
    num2 ← num2 * 7
    ESCREVA num2
    i ← i + 1
    SE i ≠ num_termos
    ENTÃO INÍCIO
        num3 ← num3 * 4
        ESCREVA num3
        i ← i + 1
    FIM
FIM
FIM
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX11_A.PAS e \EXERC\CAP4\PASCAL\EX11_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX11_B.PAS e \EXERC\CAP4\PASCAL\EX11_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX11_A.CPP e \EXERC\CAP4\C++\EX11_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX11_B.CPP e \EXERC\CAP4\C++\EX11_B.EXE

12. Faça um programa que receba o valor de X e o número de termos desejados. Calcule e mostre o valor da seguinte seqüência:

$$S = 1 + X^2/3! - X^3/4! + X^4/5! - X^5/6! + X^6/7! - \dots$$



SOLUÇÃO:

```

ALGORITMO
DECLARE i, j, x, num, s, fat NUMÉRICO
LEIA x, num
s ← 1
PARA i ← 2 ATÉ num FAÇA
INÍCIO
    fat ← 1
    PARA j ← 1 ATÉ i+1 FAÇA
        INÍCIO
            fat ← fat * j
        FIM
    SE RESTO (i/2) = 0
    ENTÃO s ← s + (xi)/fat
    SENÃO s ← s - (xi)/fat
FIM
ESCREVA s
FIM-ALGORITMO.

```


**RESOLUÇÃO
PASCAL**
1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX12_A.PAS e \EXERC\CAP4\PASCAL\EX12_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX12_B.PAS e \EXERC\CAP4\PASCAL\EX12_B.EXE


**RESOLUÇÃO
C/C++**
1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX12_A.CPP e \EXERC\CAP4\C++\EX12_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX12_B.CPP e \EXERC\CAP4\C++\EX12_B.EXE

13. Faça um programa que receba duas notas de seis alunos, calcule e mostre:

- ♦ a média aritmética das duas notas de cada aluno;
- ♦ a mensagem que está na tabela a seguir:

| MÉDIA ARITMÉTICA | MENSAGEM |
|------------------|-----------|
| Até 3,0 | Reprovado |
| Entre 3,0 e 7,0 | Exame |
| Acima de 7,0 | Aprovado |

- ♦ o total de alunos aprovados;
- ♦ o total de alunos de exame;
- ♦ o total de alunos reprovados;
- ♦ a média da classe.


ALGORITMO
SOLUÇÃO:

```

ALGORITMO
DECLARE cont, n1, n2, media, ta, te, tr NUMÉRICO
media_classe, total_classe NUMÉRICO
total_classe ← 0
PARA cont ← 1 ATÉ 6 FAÇA
    INÍCIO
        LEIA n1, n2
        media ← (n1 + n2) / 2
        ESCREVA media
        SE media <= 3
            ENTÃO INÍCIO
                tr ← tr + 1
                ESCREVA "Reprovado"
            FIM
        SE (media > 3) E (media < 7)
            ENTÃO INÍCIO
                te ← te + 1
                ESCREVA "Exame"
            FIM
        SE (media >= 7)
            ENTÃO INÍCIO
                ta ← ta + 1
                ESCREVA "Aprovado"
            FIM
    FIM

```

```

        total_classe ← total_classe + media
    FIM
    ESCRIVA tr
    ESCRIVA te
    ESCRIVA ta
    media_classe ← total_classe/6
    ESCRIVA media_classe
    FIM_ALGORITMO.

```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX13_A.PAS e \EXERC\CAP4\PASCAL\EX13_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX13_B.PAS e \EXERC\CAP4\PASCAL\EX13_B.EXE



RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX13_A.CPP e \EXERC\CAP4\C++\EX13_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX13_B.CPP e \EXERC\CAP4\C++\EX13_B.EXE

14. Faça um programa que calcule a soma dos primeiros 50 números pares. Esse programa não recebe valor do teclado. Os primeiros números pares são: 2, 4, 6,



ALGORITMO

Solução:

```

ALGORITMO
    DECLARE soma, num, qtde NUMÉRICO
    soma ← 0
    num ← 2
    PARA qtde ← 1 ATÉ 50 FAÇA
        INÍCIO
            soma ← soma + num
            num ← num + 2
        FIM
    ESCRIVA soma
    FIM_ALGORITMO.

```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX14_A.PAS e \EXERC\CAP4\PASCAL\EX14_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX14_B.PAS e \EXERC\CAP4\PASCAL\EX14_B.EXE



RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX14_A.CPP e \EXERC\CAP4\C++\EX14_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX14_B.CPP e \EXERC\CAP4\C++\EX14_B.EXE

15. Em um campeonato de futebol existem cinco times e cada time possui onze jogadores. Faça um programa que receba a idade, o peso e a altura de cada um dos jogadores, calcule e mostre:

- ◆ a quantidade de jogadores com idade inferior a 18 anos;
- ◆ a média das idades dos jogadores de cada time;
- ◆ a média das alturas de todos os jogadores do campeonato;
- ◆ a percentagem de jogadores com mais de 80 quilos entre todos os jogadores do campeonato.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
DECLARE cont_time, cont_jog, idade NUMÉRICO
peso, alt, qtde, media_idade NUMÉRICO
media_altura, porc, total80 NUMÉRCIO
qtde ← 0
total80 ← 0
PARA cont_time ← 1 ATÉ 5 FAÇA
  INÍCIO
    media_idade ← 0
    PARA cont_jog ← 1 ATÉ 11 FAÇA
      INÍCIO
        leia idade, peso, alt
        SE idade < 18
          ENTÃO qtde ← qtde + 1
        media_idade ← media_idade + idade
        media_altura ← media_altura + alt
        SE peso > 80
          ENTÃO tot80 ← tot80 + 1
      FIM
    media_idade ← media_idade/11
    ESCREVA media_idade
  FIM
ESCREVA qtde
media_altura ← media_altura/55
ESCREVA media_altura
porc ← tot80 * 100 / 55
ESCREVA porc
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX15_A.PAS e \EXERC\CAP4\PASCAL\EX15_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:


\EXERC\CAP4\PASCAL\EX15_B.PAS e \EXERC\CAP4\PASCAL\EX15_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX15_A.CPP e \EXERC\CAP4\C++\EX15_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX15_B.CPP e \EXERC\CAP4\C++\EX15_B.EXE

 **16.** Faça um programa que receba dois números. Calcule e mostre:

- ◆ a soma dos números pares desse intervalo de números, incluindo os números digitados;
- ◆ a multiplicação dos números ímpares desse intervalo de números, incluindo os números digitados.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
DECLARE num1, num2 , soma, mult, i NUMÉRICO
LEIA num1, num2
soma ← 0
mult ← 1
SE num1 = num2
ENTÃO INÍCIO
    SE RESTO (num1/2) = 0
        ENTÃO soma ← soma + num1
        SENÃO mult ← mult * num1
    FIM
SE num1 < num2
ENTÃO INÍCIO
    PARA i ← num1 ATÉ num2 FAÇA
        INÍCIO
            SE RESTO (i/2) = 0
                ENTÃO soma ← soma + i
                SENÃO mult ← mult * i
            FIM
        FIM
SE num1 > num2
ENTÃO INÍCIO
    PARA i ← num2 ATÉ num1 FAÇA
        INÍCIO
            SE RESTO (i/2) = 0
                ENTÃO soma ← soma + i
                SENÃO mult ← mult * i
            FIM
        FIM
    FIM
ESCREVA soma
ESCREVA mult
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX16_A.PAS e \EXERC\CAP4\PASCAL\EX16_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:


\EXERC\CAP4\PASCAL\EX16_B.PAS e \EXERC\CAP4\PASCAL\EX16_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX16_A.CPP e \EXERC\CAP4\C++\EX16_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX16_B.CPP e \EXERC\CAP4\C++\EX16_B.EXE

 **17.** Faça um programa que receba dois números. Calcule e mostre:

- caso os números formem um intervalo crescente, a média dos números do intervalo, incluindo os números digitados;
- caso os números formem um intervalo decrescente, a quantidade de números pares, incluindo os números digitados;
- se os números forem iguais, mostrar uma mensagem.

ALGORITMO

SOLUÇÃO:

```

ALGORITMO
DECLARE num1, num2 , soma, media, qtde, i, qtde_pares NUMÉRICO
LEIA num1, num2
soma ← 0
qtde ← 0
qtde_pares ← 0
SE num1 = num2
ENTÃO ESCREVA "Números iguais"
SE num1 < num2
ENTÃO INÍCIO
    PARA i ← num1 ATÉ num2 FAÇA
        INÍCIO
            soma ← soma + i
            qtde ← qtde + 1
        FIM
    media ← soma/qtde
    ESCREVA media
    FIM
SE num1 > num2
ENTÃO INÍCIO
    PARA i ← num2 ATÉ num1 FAÇA
        INÍCIO
            SE RESTO (i/2) = 0
                ENTÃO qtde_pares ← qtde_pares + 1
            FIM
        ESCREVA qtde_pares
    FIM
FIM_ALGORITMO.
  
```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX17_A.PAS e \EXERC\CAP4\PASCAL\EX17_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX17_B.PAS e \EXERC\CAP4\PASCAL\EX17_B.EXE




RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX17_A.CPP e \EXERC\CAP4\C++\EX17_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX17_B.CPP e \EXERC\CAP4\C++\EX17_B.EXE

 **18.** Faça um programa que receba um número inteiro maior que 1 e verifique se o número fornecido é primo ou não. Mostrar mensagem de número primo ou de número não primo.

OBSERVAÇÃO:

Um número é primo quando é divisível *apenas* pelo número 1 e por ele mesmo.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
    DECLARE i, num, qtde NUMÉRICO
    LEIA num
    qtde ← 0
    PARA i ← 1 ATÉ num FAÇA
        INÍCIO
            SE RESTO(num/i) = 0
                ENTÃO qtde ← qtde + 1
            FIM
    SE qtde > 2
        ENTÃO ESCREVA "Número não primo"
    SENÃO ESCREVA "Número primo"
    FIM_ALGORITMO.

```

**PASCAL****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\PASCAL\EX18_A.PAS e \EXERC\CAP4\PASCAL\EX18_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX18_B.PAS e \EXERC\CAP4\PASCAL\EX18_B.EXE

**C/C++****1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX18_A.CPP e \EXERC\CAP4\C++\EX18_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX18_B.CPP e \EXERC\CAP4\C++\EX18_B.EXE

19. Tem-se um conjunto de dados contendo a altura e o sexo (M ou F) de 15 pessoas. Faça um programa que calcule e mostre:

- a maior e a menor altura do grupo;
- a média de altura das mulheres;
- o número de homens;
- o sexo da pessoa mais alta.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
    DECLARE altura, cont, maior, menor, media NUMÉRICO
    soma, mediaf, somaf, cf, cm NUMÉRICO
    sexo, maior_sexo LITERAL
    soma ← 0
    somaf ← 0
    cf ← 0
    cm ← 0
    PARA cont ← 1 ATÉ 15 FAÇA
        INÍCIO
            LEIA altura, sexo
            SE cont = 1
                ENTÃO INÍCIO
                    maior ← altura
                    maior_sexo ← sexo
                    menor ← altura
                FIM
            soma ← soma + altura
            SE sexo = 'F'
                somaf ← somaf + altura
                cf ← cf + 1
            SE sexo = 'M'
                cm ← cm + 1
        FIM
    media ← soma / 15
    mediaf ← somaf / cf
    maior ← maior
    menor ← menor
    maior_sexo ← maior_sexo
    FIM

```

```

SENÃO INÍCIO
    SE altura > maior
        ENTÃO INÍCIO
            maior ← altura
            maior_sexo ← sexo
        FIM
    SE altura < menor
        ENTÃO menor ← altura
    FIM
SE sexo = "F"
    ENTÃO INÍCIO
        somaf ← somaf + altura
        cf ← cf + 1
    FIM
SENÃO cm ← cm + 1
FIM
ESCREVA maior, menor
SE cf = 0
    ENTÃO mediaf ← 0
SENÃO mediaf ← somaf/cf
ESCREVA mediaf
ESCREVA cm
ESCREVA maior_sexo
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX19_A.PAS e \EXERC\CAP4\PASCAL\EX19_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX19_B.PAS e \EXERC\CAP4\PASCAL\EX19_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX19_A.CPP e \EXERC\CAP4\C++\EX19_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX19_B.CPP e \EXERC\CAP4\C++\EX19_B.EXE

20. A conversão de graus Fahrenheit para Celsius é obtida por $c = 5/9 * (f - 32)$. Faça um programa que calcule e escreva uma tabela de graus Celsius e graus Fahrenheit, cujos graus variem de 50 a 65 de 1 em 1.



SOLUÇÃO:

```

ALGORITMO
DECLARE cels, faren NUMÉRICO
PARA faren ← 50 ATÉ 65 FAÇA
    INÍCIO
        ESCREVA faren
        cels ← 5/9 * (faren - 32)
        ESCREVA cels
    FIM
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX20_A.PAS e \EXERC\CAP4\PASCAL\EX20_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX20_B.PAS e \EXERC\CAP4\PASCAL\EX20_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:**

\EXERC\CAP4\C++\EX20_A.CPP e \EXERC\CAP4\C++\EX20_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX20_B.CPP e \EXERC\CAP4\C++\EX20_B.EXE

21. Em uma fábrica trabalham homens e mulheres divididos em três classes:

- ◆ trabalhadores que fazem até 30 peças por mês – classe 1;
- ◆ trabalhadores que fazem de 31 a 35 peças por mês – classe 2;
- ◆ trabalhadores que fazem mais de 35 peças por mês – classe 3.

A classe 1 recebe salário mínimo. A classe 2 recebe salário mínimo mais 3% do salário mínimo por peça, acima das 30 peças iniciais. A classe 3 recebe salário mínimo mais 5% do salário mínimo por peça, acima das 30 peças iniciais.

Faça um programa que receba o número do operário, o número de peças fabricadas no mês, o sexo do operário, e que também calcule e mostre:

- ◆ o número do operário e seu salário;
- ◆ o total da folha de pagamento da fábrica;
- ◆ o número total de peças fabricadas no mês;
- ◆ a média de peças fabricadas pelos homens;
- ◆ a média de peças fabricadas pelas mulheres;
- ◆ o número do operário ou operária de maior salário.

A fábrica possui 15 operários.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
DECLARE num_op, pecas_op, num_maior, cont_m, cont_f NUMÉRICO
tot_pecas, cont, media_m NUMÉRICO
    media_f, salario_op, tot_folha NUMÉRICO
    sexo_op LITERAL
tot_folha ← 0
tot_pecas ← 0
media_m ← 0
media_f ← 0
cont_m ← 0
cont_f ← 0
PARA cont ← 1 ATÉ 15 FAÇA
    INÍCIO
        ESCREVA "Digite a número do ", cont, "º operário "
        LEIA num_op
        ESCREVA "Digite o sexo do operário (M ou F) "
        LEIA sexo_op
        ESCREVA "Digite o total de peças fabricadas pelo ", cont, "º
            operário "
        LEIA pecas_op
        SE pecas_op <= 30
            ENTÃO salario_op ← 150
        SE (pecas_op > 30) E (pecas_op <= 35)

```

```

        ENTÃO salario_op ← 150 + ((pecas_op - 30) * 3 / 100 * 150)
SE pecas_op > 35
    ENTÃO salario_op ← 150 + ((pecas_op - 35) * 5 / 100 * 150)
    ESCRIVA "O operário de número ", num_op, " recebe salário = ",
    ■ salario_op
    tot_folha ← tot_folha + salario_op
    tot_pecas ← tot_pecas + pecas_op
SE sexo_op = 'M'
    ENTÃO INÍCIO
        media_m ← media_m + pecas_op
        cont_m ← cont_m + 1
    FIM
    SENÃO INÍCIO
        media_f ← media_f + pecas_op
        cont_f ← cont_f + 1
    FIM
SE cont = 1
    ENTÃO INÍCIO
        salario_maior ← salario_op
        num_maior ← num_op
    FIM
    SENÃO INÍCIO
        SE (salario_op > salario_maior)
            ENTÃO INÍCIO
                salario_maior ← salario_op
                num_maior ← num_op
            FIM
        FIM
FIM
FIM
ESCREVA "Total da folha de pagamento = ", tot_folha
ESCREVA "Total de peças fabricadas no mês = ", tot_pecas
SE cont_m = 0
    ENTÃO media_m ← 0
    SENÃO media_m ← media_m / cont_m
SE cont_f = 0
    ENTÃO media_f ← 0
    SENÃO media_f ← media_f / cont_f
ESCREVA "Média de peças fabricadas por mulheres = ", media_f
ESCREVA "Média de peças fabricadas por homens = ", media_m
ESCREVA "O número do operário com maior salário é ", num_maior
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX21_A.PAS e \EXERC\CAP4\PASCAL\EX21_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX21_B.PAS e \EXERC\CAP4\PASCAL\EX21_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX21_A.CPP e \EXERC\CAP4\C++\EX21_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX21_B.CPP e \EXERC\CAP4\C++\EX21_B.EXE

22. Foi feita uma pesquisa para determinar o índice de mortalidade infantil em um certo período. Faça um programa que:

- ◆ leia o número de crianças nascidas no período;
- ◆ o sexo (M ou F) e o tempo de vida para cada criança nascida.

Calcule e mostre:

- ◆ a percentagem de crianças do sexo feminino mortas no período;
- ◆ a percentagem de crianças do sexo masculino mortas no período;
- ◆ a percentagem de crianças que viveram 24 meses ou menos no período.

ALGORITMO

Solução:

```

ALGORITMO
DECLARE i, num_cri, meses, porc_f, porc_m, tot_f, NUMÉRICO
        tot_m, tot24, porc24 NUMÉRICO
        sexo LITERAL
ESCREVA "Digite o número de crianças nascidas no período "
LEIA num_cri
tot_m ← 0
tot_f ← 0
tot24 ← 0
PARA i=1 ATE num_cri FAÇA
    INÍCIO
        ESCREVA "Digite o sexo da ", i, "ª criança"
        LEIA sexo
        ESCREVA "Digite o tempo de vida (em meses) da ", i, "ª
        ➤ criança"
        LEIA meses
        SE sexo = "M"
            ENTÃO tot_m ← tot_m + 1
        SE sexo = "F"
            ENTÃO tot_f ← tot_f + 1
        SE meses <= 24
            ENTÃO tot_24 ← tot_24 + 1
    FIM
SE num_cri = 0
    ENTÃO INÍCIO
        perc_m ← 0
        perc_f ← 0
        perc_24 ← 0
    FIM
SENÃO INÍCIO
        perc_m ← tot_m * 100 / num_cri
        perc_f ← tot_f * 100 / num_cri
        perc_24 ← tot_24 * 100 / num_cri
    FIM
ESCREVA "Percentual de crianças do sexo feminino mortas ", porc_f
ESCREVA "Percentual de crianças do sexo masculino mortas ", porc_m
ESCREVA "Percentual de crianças com 24 meses ou menos mortas no
➤ período ", porc_24
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX22_A.PAS e \EXERC\CAP4\PASCAL\EX22_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX22_B.PAS e \EXERC\CAP4\PASCAL\EX22_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX22_A.CPP e \EXERC\CAP4\C++\EX22_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX22_B.CPP e \EXERC\CAP4\C++\EX22_B.EXE

23. Faça um programa que receba o valor de uma dívida e mostre uma tabela com os seguintes dados: valor da dívida, valor dos juros, quantidade de parcelas e valor da parcela.

Os juros e a quantidade de parcelas seguem a tabela abaixo:

| QUANTIDADE DE PARCELAS | % DE JUROS SOBRE O VALOR INICIAL DA DÍVIDA |
|---------------------------|---|
| 1 | 0 |
| 3 | 10 |
| 6 | 15 |
| 9 | 20 |
| 12 | 25 |

Exemplo de saída do programa:

| VALOR DA DÍVIDA | VALOR DOS JUROS | QUANTIDADE DE PARCELAS | VALOR DA PARCELA |
|--------------------|--------------------|---------------------------|---------------------|
| R\$ 1.000,00 | 0 | 1 | R\$ 1.000,00 |
| R\$ 1.100,00 | 100 | 3 | R\$ 366,67 |
| R\$ 1.150,00 | 150 | 6 | R\$ 191,67 |

ALGORITMO

SOLUÇÃO:

```

ALGORITMO
DECLARE valor_inicial, juros, valor_parco NUMÉRICO
total, valor_juros, num_parco, i NUMÉRICO
ESCREVA "Digite o valor inicial da dívida"
LEIA valor_inicial
juros ← 0
num_parco ← 1
total ← valor_inicial
valor_parco ← valor_inicial
ESCREVA total
ESCREVA juros
ESCREVA num_parco
ESCREVA valor_parco
juros ← juros + 10
num_parco ← num_parco + 2
PARA i ← 1 ATÉ 4 FAÇA
  INÍCIO
    valor_juros ← valor_inicial * juros / 100
    total ← valor_inicial + valor_juros
    valor_parco ← total / num_parco
    ESCREVA total
    ESCREVA valor_juros
    ESCREVA num_parco
    ESCREVA valor_parco
    juros ← juros + 5
    num_parco ← num_parco + 3
  FIM
FIM
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

```
\EXERC\CAP4\PASCAL\EX23_A.PAS e \EXERC\CAP4\PASCAL\EX23_A.EXE
```

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

```
\EXERC\CAP4\PASCAL\EX23_B.PAS e \EXERC\CAP4\PASCAL\EX23_B.EXE
```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

```
\EXERC\CAP4\C++\EX23_A.CPP e \EXERC\CAP4\C++\EX23_A.EXE
```

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

```
\EXERC\CAP4\C++\EX23_B.CPP e \EXERC\CAP4\C++\EX23_B.EXE
```

24. Faça um programa que receba o preço unitário, a refrigeração (S para os produtos que necessitem de refrigeração e N para os produtos que não necessitem de refrigeração) e a categoria (A – Alimentação, L – Limpeza e V – Vestuário) de 12 produtos. Calcule e mostre:

- o custo de estocagem calculado de acordo com a tabela a seguir.

| PREÇO UNITÁRIO | REFRIGERAÇÃO | CATEGORIA | CUSTO DE ESTOCAGEM |
|--|--------------|-----------|--------------------|
| Até R\$ 20,00 | | A | R\$ 2,00 |
| | | L | R\$ 3,00 |
| | | V | R\$ 4,00 |
| Entre R\$ 20,00 e R\$ 50,00 (inclusive) | S | | R\$ 6,00 |
| | N | | R\$ 0,00 |
| Maior que R\$ 50,01 | S | A | R\$ 5,00 |
| | | L | R\$ 2,00 |
| | | V | R\$ 4,00 |
| | N | A ou V | R\$ 0,00 |
| | | L | R\$ 1,00 |

- o imposto calculado de acordo com as regras a seguir:

Se o produto **não preencher** nenhum dos requisitos abaixo, seu imposto será de 2% sobre o preço unitário; caso contrário, será de 4%.

Os requisitos são: Categoria – A e Refrigeração – S

- o preço final calculado observando as informações anteriores;
- a classificação calculada usando a tabela a seguir.

| PREÇO FINAL | CLASSIFICAÇÃO |
|------------------------------|---------------|
| Até R\$ 20,00 | Barato |
| Entre R\$ 20,00 e R\$ 100,00 | Normal |
| Acima de R\$ 100,00 | Caro |

- a média dos valores adicionais;

- ♦ o maior preço final;
- ♦ o menor preço final;
- ♦ o total dos impostos;
- ♦ a quantidade de produtos com classificação Barato;
- ♦ a quantidade de produtos com classificação Caro;
- ♦ a quantidade de produtos com classificação Normal.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
DECLARE i, preco, custo_est, imp, preco_final, adicional NUMÉRICO
        maior_p, menor_p, tot_imp, qtd_b, qtd_n, qtd_c NUMÉRICO
        refri, categ LITERAL
adicional ← 0
tot_imp ← 0
qtd_b ← 0
qtd_n ← 0
qtd_c ← 0
PARA i ← 1 ATÉ 12 FAÇA
INÍCIO
    LEIA preco
    LEIA refri
    LEIA categ
    SE (preco ≤ 20)
        ENTÃO INÍCIO
            SE (categ = "A")
                ENTÃO custo_est ← 2
            SE (categ = "L")
                ENTÃO custo_est ← 3
            SE (categ = "V")
                ENTÃO custo_est ← 4
        FIM
    SE (preco > 20) E (preco ≤ 50)
        ENTÃO INÍCIO
            SE (refri = "S")
                ENTÃO custo_est ← 6
            SENÃO custo_est ← 0
        FIM
    SE (preco > 50)
        ENTÃO INÍCIO
            SE (refri = "S")
                ENTÃO INÍCIO
                    SE (categ = "A")
                        ENTÃO custo_est ← 5
                    SE (categ = "L")
                        ENTÃO custo_est ← 2
                    SE (categ = "V")
                        ENTÃO custo_est ← 4
                FIM
            SENÃO INÍCIO
                SE (categ = "A") OU (categ = "V")
                    ENTÃO custo_est ← 0
                SE (categ = "L")
                    ENTÃO custo_est ← 1
            FIM
        FIM
    SE (categ ≠ "A") E (refri ≠ "S")
        ENTÃO imp ← preco * 2 / 100
        SENÃO imp ← preco * 4 / 100
    preco_final ← preco + custo_est + imp
    ESCRIVA preco

```



```

    ESCRIVA custo_est
    ESCRIVA imp
    ESCRIVA preco_final
    SE (preco_final <= 20)
        ENTÃO INÍCIO
            qtd_b ← qtd_b + 1
            ESCRIVA "Classificação Barato"
        FIM
    SE (preco_final > 20) E (preco_final < 100)
        ENTÃO INÍCIO
            qtd_n ← qtd_n + 1
            ESCRIVA "Classificação Normal"
        FIM
    SE (preco_final > 100)
        ENTÃO INÍCIO
            qtd_c ← qtd_c + 1
            ESCRIVA "Classificação Caro"
        FIM
    adicional ← adicional + custo_est + imp
    tot_imp ← tot_imp + imp
    SE (i = 1)
        ENTÃO INÍCIO
            Maior_p ← preco_final
            Menor_p ← preco_final
        FIM
    SENÃO INÍCIO
        SE (preco_final > maior_p)
            ENTÃO maior_p ← preco_final
        SE (preco_final < menor_p)
            ENTÃO menor_p ← preco_final
        FIM
    FIM
    adicional ← adicional / 12
    ESCRIVA adicional
    ESCRIVA maior_p
    ESCRIVA menor_p
    ESCRIVA tot_imp
    ESCRIVA qtd_b
    ESCRIVA qtd_n
    ESCRIVA qtd_c
    FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX24_A.PAS e \EXERC\CAP4\PASCAL\EX24_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX24_B.PAS e \EXERC\CAP4\PASCAL\EX24_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX24_A.CPP e \EXERC\CAP4\C++\EX24_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX24_B.CPP e \EXERC\CAP4\C++\EX24_B.EXE

25. Faça um programa que leia uma quantidade indeterminada de números positivos e conte quantos deles estão nos seguintes intervalos: [0-25], [26-50], [51-75] e [76-100]. A entrada de dados deverá terminar quando for lido um número negativo.

ALGORITMO**Solução:**

```

ALGORITMO
DECLARE num, int1, int2, int3, int4 NUMÉRICO
int1 ← 0
int2 ← 0
int3 ← 0
LEIA num
ENQUANTO (num >= 0) FAÇA
INÍCIO
    SE (num >= 0) E (num <= 25)
        ENTÃO int1 ← int1 + 1
    SE (num >= 26) E (num <= 50)
        ENTÃO int2 ← int2 + 1
    SE (num >= 51) E (num <= 75)
        ENTÃO int3 ← int3 + 1
    SE (num >= 76) E (num <= 100)
        ENTÃO int4 ← int4 + 1
    LEIA num
FIM
ESCREVA int1
ESCREVA int2
ESCREVA int3
ESCREVA int4
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\PASCAL\EX25_A.PAS e \EXERC\CAP4\PASCAL\EX25_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX25_B.PAS e \EXERC\CAP4\PASCAL\EX25_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\C++\EX25_A.CPP e \EXERC\CAP4\C++\EX25_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX25_B.CPP e \EXERC\CAP4\C++\EX25_B.EXE

26. Faça um programa que determine e mostre os cinco primeiros múltiplos de 3, considerando números > 0.

ALGORITMO**Solução:**

```

ALGORITMO
DECLARE qtd, num NUMÉRICO
qtd ← 0
num ← 1
ENQUANTO (qtd < 5) FAÇA
INÍCIO
    SE (RESTO(num/3) = 0)
        ENTÃO INÍCIO

```

```

        ESCREVA num
        qtd ← qtd + 1
    FIM
    num ← num + 1
FIM
FIM_ALGORITMO.

```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX26_A.PAS e \EXERC\CAP4\PASCAL\EX26_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX26_B.PAS e \EXERC\CAP4\PASCAL\EX26_B.EXE



RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX26_A.CPP e \EXERC\CAP4\C++\EX26_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX26_B.CPP e \EXERC\CAP4\C++\EX26_B.EXE

27. Faça um programa para calcular a área de um triângulo. Esse programa não pode permitir a entrada de dados inválidos, ou seja, medidas menores ou iguais a 0.



ALGORITMO

SOLUÇÃO:

```

ALGORITMO
DECLARE base, altura, area NUMÉRICO
REPITA
    LEIA base
    ATÉ QUE (base > 0)
    REPITA
        LEIA altura
        ATÉ QUE (altura > 0)
    area ← base * altura / 2
    ESCREVA area
FIM_ALGORITMO

```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX27_A.PAS e \EXERC\CAP4\PASCAL\EX27_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX27_B.PAS e \EXERC\CAP4\PASCAL\EX27_B.EXE



RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX27_A.CPP e \EXERC\CAP4\C++\EX27_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX27_B.CPP e \EXERC\CAP4\C++\EX27_B.EXE

28. O cardápio de uma lanchonete é o seguinte:

| ESPECIFICAÇÃO | CÓDIGO | PREÇO |
|-----------------|--------|----------|
| Cachorro quente | 100 | R\$ 1,20 |
| Bauru simples | 101 | R\$ 1,30 |
| Bauru com ovo | 102 | R\$ 1,50 |
| Hambúrguer | 103 | R\$ 1,20 |
| Cheeseburger | 104 | R\$ 1,30 |
| Refrigerante | 105 | R\$ 1,00 |

Faça um programa que leia o código dos itens pedidos e as quantidades desejadas. Calcule e mostre o valor a ser pago por item (preço * quantidade) e o total geral do pedido. Considere que o cliente deve informar quando o pedido deve ser encerrado.

ALGORITMO

SOLUÇÃO:

```

ALGORITMO
DECLARE codigo, qtd, valor_item, valor_total NUMÉRICO
      Resposta LITERAL
valor_total ← 0
REPITA
    LEIA codigo
    LEIA qtd
    SE (codigo = 100)
        ENTÃO INÍCIO
            ESCREVA "Cachorro Quente"
            valor_item ← qtd * 1,20
        FIM
    SE (codigo = 101)
        ENTÃO INÍCIO
            ESCREVA "Bauru Simples"
            valor_item ← qtd * 1,30
        FIM
    SE (codigo = 102)
        ENTÃO INÍCIO
            ESCREVA "Bauru com Ovo"
            valor_item ← qtd * 1,50
        FIM
    SE (codigo = 103)
        ENTÃO INÍCIO
            ESCREVA "Hambúrguer"
            valor_item ← qtd * 1,20
        FIM
    SE (codigo = 104)
        ENTÃO INÍCIO
            ESCREVA "Cheeseburger"
            valor_item ← qtd * 1,30
        FIM
    SE (codigo = 105)
        ENTÃO INÍCIO
            ESCREVA "Refrigerante"
            valor_item ← qtd * 1,0
        FIM
    ESCREVA valor_item
    valor_total ← valor_total + valor_item
    ESCREVA "Deseja mais alguma coisa (S ou N) ? "
    LEIA resp
    ATÉ resp = 'N'
    ESCREVA valor_total
FIM_ALGORITMO.

```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX28_A.PAS e \EXERC\CAP4\PASCAL\EX28_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX28_B.PAS e \EXERC\CAP4\PASCAL\EX28_B.EXE



RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX28_A.CPP e \EXERC\CAP4\C++\EX28_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX28_B.CPP e \EXERC\CAP4\C++\EX28_B.EXE

29. Faça um programa que receba o salário de um funcionário chamado Carlos. Sabe-se que o funcionário João tem um salário equivalente a um terço do salário de Carlos. Carlos aplicará seu salário integralmente na caderneta de poupança, que está rendendo 2% ao mês e João aplicará seu salário integralmente no fundo de renda fixa, que está rendendo 5% ao mês. Calcule e mostre a quantidade de meses necessários para que o valor pertencente a João iguale ou ultrapasse o valor pertencente a Carlos.

ALGORITMO

SOLUÇÃO:

```
ALGORITMO
DECLARE sal_carlos, sal_joao, meses NUMÉRICO
LEIA sal_carlos
sal_joao ← sal_carlos / 3
meses ← 0
ENQUANTO (sal_joao < sal_carlos) FAÇA
INÍCIO
    sal_carlos ← sal_carlos + (sal_carlos * 2 / 100)
    sal_joao ← sal_joao + (sal_joao * 5 / 100)
    meses ← meses + 1
FIM
ESCREVA meses
FIM_ALGORITMO.
```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX29_A.PAS e \EXERC\CAP4\PASCAL\EX29_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX29_B.PAS e \EXERC\CAP4\PASCAL\EX29_B.EXE



RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX29_A.CPP e \EXERC\CAP4\C++\EX29_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX29_B.CPP e \EXERC\CAP4\C++\EX29_B.EXE

30. Faça um programa que leia um conjunto não determinado de valores, um de cada vez, e escreva uma tabela com cabeçalho, que deve ser repetido a cada 20 linhas. A tabela deverá conter o valor lido, seu quadrado, seu cubo e sua raiz quadrada. Finalizar a entrada de dados com um valor negativo ou zero.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
DECLARE linhas, num, quad, cubo, raiz NUMÉRICO
LEIA num
ESCREVA "Valor   Quadrado   Cubo   Raiz"
linhas ← 1
ENQUANTO (num >= 0) FAÇA
INÍCIO
quad ← num * num
cubo ← num * num * num
raiz ←  $\sqrt{\text{num}}$ 
SE (linhas < 20)
ENTÃO INÍCIO
    linhas ← linhas + 1
    ESCRIVA (quad, cubo, raiz)
FIM
SENÃO INÍCIO
    LIMPAR A TELA
    linhas ← 1
    ESCRIVA "Valor   Quadrado   Cubo   Raiz"
    linhas ← linhas + 1
    ESCRIVA quad, cubo, raiz
FIM
LEIA num
FIM
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\PASCAL\EX30_A.PAS e \EXERC\CAP4\PASCAL\EX30_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX30_B.PAS e \EXERC\CAP4\PASCAL\EX30_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\C++\EX30_A.CPP e \EXERC\CAP4\C++\EX30_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX30_B.CPP e \EXERC\CAP4\C++\EX30_B.EXE

31. Faça um programa que leia um número não determinado de pares de valores [m,n], todos inteiros e positivos, um par de cada vez e que calcule e mostre a soma de todos os números inteiros entre m e n (inclusive). A digitação de pares termina quando m for maior ou igual a n.

ALGORITMO**SOLUÇÃO:**

ALGORITMO

```

DECLARE m, n, soma, i NUMÉRICO
LEIA m
LEIA n
ENQUANTO (m < n) FAÇA
INÍCIO
    soma ← 0
    PARA i = m ATÉ n FAÇA
    INÍCIO
        soma ← soma + i
    FIM
    ESCREVA soma
    LEIA m
    LEIA n
FIM
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX31_A.PAS e \EXERC\CAP4\PASCAL\EX31_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX31_B.PAS e \EXERC\CAP4\PASCAL\EX31_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX31_A.CPP e \EXERC\CAP4\C++\EX31_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX31_B.CPP e \EXERC\CAP4\C++\EX31_B.EXE

32. Faça um programa que leia dois valores inteiros e positivos, X e Y, e que calcule e mostre a potência X^Y , utilizando uma estrutura de repetição.



SOLUÇÃO:

```

ALGORITMO
DECLARE x, y, pot, cont NUMÉRICO
LEIA x
LEIA y
pot ← 1
PARA cont ← 1 ATÉ y FAÇA
INÍCIO
    pot ← pot * x
FIM
ESCREVA pot
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\PASCAL\EX32_A.PAS e \EXERC\CAP4\PASCAL\EX32_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX32_B.PAS e \EXERC\CAP4\PASCAL\EX32_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA FOR:

\EXERC\CAP4\C++\EX32_A.CPP e \EXERC\CAP4\C++\EX32_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX32_B.CPP e \EXERC\CAP4\C++\EX32_B.EXE

33. Faça um programa para ler o código, o sexo (M – Masculino, F – Feminino) e o número de horas/aula dadas mensalmente pelos professores de uma universidade, sabendo-se que cada hora/aula vale R\$ 18,50. Emita uma listagem contendo o código, o salário bruto e o salário líquido (levando em consideração os descontos explicados a seguir) de todos os professores lidos. Mostre também a média dos salários líquidos dos professores do sexo masculino e a média dos salários brutos dos professores do sexo feminino. Considere:

- ♦ desconto para homens 10% e para mulheres 5%;
- ♦ as informações terminarão quando for lido o código = 99999.

ALGORITMO

SOLUÇÃO:

```

ALGORITMO
DECLARE cod, num_h, sal_b, sal_l, media_m, media_f NUMÉRICO
        cont_m, cont_f NUMÉRICO
        Sexo LITERAL
LEIA cod
cont_m ← 0
cont_f ← 0
ENQUANTO (cod ≠ 99999) FAÇA
INÍCIO
    LEIA sexo
    LEIA num_h
    sal_b ← num_h * 18,50
    SE (sexo = "M")
        ENTÃO INÍCIO
            sal_l ← sal_b - (sal_b * 10 / 100)
            media_m ← media_m + sal_l
            cont_m ← cont_m + 1
        FIM
    SE (sexo = "F")
        ENTÃO INÍCIO
            sal_l ← sal_b - (sal_b * 5 / 100)
            media_f ← media_f + sal_l
            cont_f ← cont_f + 1
        FIM
    ESCRIVA cod
    ESCRIVA sal_b
    ESCRIVA sal_l
    LEIA cod
FIM
media_m ← media_m / cont_m
media_f ← media_f / cont_f
ESCREVA media_m
ESCREVA media_f
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX33_A.PAS e \EXERC\CAP4\PASCAL\EX33_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX33_B.PAS e \EXERC\CAP4\PASCAL\EX33_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\C++\EX33_A.CPP e \EXERC\CAP4\C++\EX33_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX33_B.CPP e \EXERC\CAP4\C++\EX33_B.EXE

34. Faça um programa que leia um número indeterminado de valores para m, todos inteiros e positivos, um de cada vez. Se m for par, verifique quantos divisores possui. Se m for ímpar, calcule a soma dos números inteiros de 1 até m (m não deve entrar nos cálculos). Mostre os cálculos realizados. Finalize a entrada de dados com m zero ou negativo.

**SOLUÇÃO:**

```

ALGORITMO
DECLARE m, calc, i NUMÉRICO
LEIA m
ENQUANTO (m > 0) FAÇA
  INÍCIO
    calc ← 0
    SE (RESTO(m/2) = 0)
      ENTÃO INÍCIO
        PARA i ← 1 ATÉ m FAÇA
          INÍCIO
            SE (RESTO(m/i) = 0)
              ENTÃO calc ← calc + 1
          FIM
        ESCREVA calc
      FIM
    SENÃO INÍCIO
      PARA i ← 1 ATÉ m-1 FAÇA
        INÍCIO
          calc ← calc + i
        FIM
      ESCREVA calc
    FIM
  LEIA m
FIM
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\PASCAL\EX34_A.PAS e \EXERC\CAP4\PASCAL\EX34_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX34_B.PAS e \EXERC\CAP4\PASCAL\EX34_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\PASCAL\EX34_A.CPP e \EXERC\CAP4\PASCAL\EX34_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX34_B.CPP e \EXERC\CAP4\C++\EX34_B.EXE

35. Faça um programa que receba vários números, calcule e mostre:

- ♦ a soma dos números digitados;
- ♦ a quantidade de números digitados;
- ♦ a média dos números digitados;
- ♦ o maior número digitado;
- ♦ o menor número digitado;
- ♦ a média dos números pares;
- ♦ a percentagem dos números ímpares entre todos os números digitados.

Finalize a entrada de dados com a digitação do número 30000.

ALGORITMO**SOLUÇÃO:**

```

ALGORITMO
DECLARE num, soma, qtd, maior, menor, qtd_par NUMÉRICO
        soma_par, qtd_impar, media, perc NUMÉRICO
qtd ← 0
qtd_par ← 0
soma_par ← 0
qtd_impar ← 0
LEIA num
ENQUANTO (num <= 30000) FAÇA
INÍCIO
    SE (qtd = 0)
        ENTÃO INÍCIO
            maior ← num
            menor ← num
        FIM
    SENÃO INÍCIO
        SE (num > maior)
            ENTÃO maior ← num
        SE (num < menor)
            ENTÃO menor ← num
        FIM
    soma ← soma + num
    qtd ← qtd + 1
    SE (RESTO(num/2) = 0)
        ENTÃO INÍCIO
            soma_par ← soma_par + num
            qtd_par ← qtd_par + 1
        FIM
    SENÃO qtd_impar ← qtd_impar + 1
LEIA num
FIM
ESCREVA soma
ESCREVA qtd
media ← soma / qtd
ESCREVA media
ESCREVA maior
ESCREVA menor
media ← soma_par / qtd_par
ESCREVA media
perc ← qtd_impar * 100 / qtd
ESCREVA perc
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\PASCAL\EX35_A.PAS e \EXERC\CAP4\PASCAL\EX35_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX35_B.PAS e \EXERC\CAP4\PASCAL\EX35_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\C++\EX35_A.CPP e \EXERC\CAP4\C++\EX35_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX35_B.CPP e \EXERC\CAP4\C++\EX35_B.EXE

36. Faça um programa que:

- ◆ leia um número indeterminado de linhas contendo cada uma a idade de um indivíduo. A última linha, que não entrará nos cálculos, contém o valor da idade igual a zero;
- ◆ calcule e mostre a idade média desse grupo de indivíduos.

**SOLUÇÃO:**

```
ALGORITMO
DECLARE idade, soma, qtd, media NUMÉRICO
soma ← 0
qtd ← 0
LEIA idade
ENQUANTO (idade > 0) FAÇA
INÍCIO
    soma ← soma + idade
    qtd ← qtd + 1
    LEIA idade
FIM
media ← soma / qtd
ESCREVA soma
FIM_ALGORITMO.
```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\PASCAL\EX36_A.PAS e \EXERC\CAP4\PASCAL\EX36_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX36_B.PAS e \EXERC\CAP4\PASCAL\EX36_B.EXE

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

\EXERC\CAP4\C++\EX36_A.CPP e \EXERC\CAP4\C++\EX36_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX36_B.CPP e \EXERC\CAP4\C++\EX36_B.EXE

37. Uma empresa decidiu fazer um levantamento em relação aos candidatos que se apresentarem para preenchimento de vagas no seu quadro de funcionários. Supondo que você seja o programador dessa empresa, faça um programa que:

- ♦ leia, para cada candidato, a idade, o sexo (M ou F) e a experiência no serviço (S ou N). Para encerrar a entrada de dados digite zero para a idade.

Calcule e mostre:

- ♦ o número de candidatos do sexo feminino;
- ♦ o número de candidatos do sexo masculino;
- ♦ a idade média dos homens que já têm experiência no serviço;
- ♦ a percentagem dos homens com mais de 45 anos entre o total dos homens;
- ♦ o número de mulheres com idade inferior a 35 anos e com experiência no serviço;
- ♦ a menor idade entre as mulheres que já têm experiência no serviço.

ALGORITMO

SOLUÇÃO:

```

ALGORITMO
DECLARE idade, tot_f, tot_m, somal, cont_m1, cont_m2, cont_m3,
        cont_m4, cont_f1, media_idade, calc NUMÉRICO
        sexo, exp LITERAL

tot_f ← 0
tot_m ← 0
somal ← 0
cont_m1 ← 0
cont_m2 ← 0
cont_f1 ← 0
LEIA idade
ENQUANTO (idade ≠ 0) FAÇA
INÍCIO
    LEIA sexo
    LEIA exp
    SE (sexo = "F") E (exp = "S")
        ENTÃO INÍCIO
            SE (tot_f = 0)
                ENTÃO menor_idade ← idade
            SENÃO SE (idade < menor_idade)
                ENTÃO Menor_idade ← idade
        FIM
    SE (sexo = "M")
        ENTÃO tot_m ← tot_m + 1
    SE (sexo = "F")
        ENTÃO tot_f ← tot_f + 1
    SE (sexo = "F") E (idade < 35) E (exp = "S")
        ENTÃO cont_f1 ← cont_f1 + 1
    SE (sexo = "M") e (idade > 45)
        ENTÃO cont_m1 ← cont_m1 + 1
    SE (sexo = "M") E (exp = "S")
        ENTÃO INÍCIO
            somal ← somal + idade
            cont_m2 ← cont_m2 + 1
        FIM
    LEIA idade
FIM
ESCREVA tot_f
ESCREVA tot_m
calc ← somal / cont_m2
ESCREVA calc
Calc ← cont1 * 100 / tot_m
ESCREVA calc

```

```

ESCREVA cont_f1
ESCREVA menor_idade
FIM_ALGORITMO.

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

```

\EXERC\CAP4\PASCAL\EX37_A.PAS e \EXERC\CAP4\PASCAL\EX37_A.EXE

```

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

```

\EXERC\CAP4\PASCAL\EX37_B.PAS e \EXERC\CAP4\PASCAL\EX37_B.EXE

```

**1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:**

```

\EXERC\CAP4\C++\EX37_A.CPP e \EXERC\CAP4\C++\EX37_A.EXE

```

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

```

\EXERC\CAP4\C++\EX37_B.CPP e \EXERC\CAP4\C++\EX37_B.EXE

```

38. Faça um programa que receba o valor do salário mínimo e uma lista contendo a quantidade de quilowatts gasta por consumidor e o tipo de consumidor (1 – Residencial, 2 – Comercial ou 3 – Industrial).

Calcule e mostre:

- ♦ o valor de cada quilowatt, sabendo que o quilowatt custa $\frac{1}{8}$ do salário mínimo;
- ♦ o valor a ser pago por cada consumidor (conta final mais acréscimo), considerando que o acréscimo é o mesmo da tabela a seguir.

| TIPO | % DE ACRÉSCIMO SOBRE O VALOR GASTO |
|------|------------------------------------|
| 1 | 5 |
| 2 | 10 |
| 3 | 15 |

- ♦ o faturamento geral da empresa;
- ♦ a quantidade de consumidores que pagam entre R\$ 500,00 e R\$ 1.000,00.

Termine a entrada de dados com quantidade de quilowatts igual a zero.

**SOLUÇÃO:**

```

ALGORITMO
DECLARE sal, qtd, tipo, valor_kw, gasto, acresc NUMÉRICO
        total, tot_geral, qtd_cons NUMÉRICO
tot_geral ← 0
qtd_cons ← 0
LEIA (sal)
valor_kw ← sal / 8
ENQUANTO (qtd ≠ 0) FAÇA
INÍCIO
    gasto ← qtd * valor_kw
    LEIA tipo
    SE (tipo = 1)
        ENTÃO acresc ← gasto * 5 / 100
    SE (tipo = 2)
        ENTÃO acresc ← gasto * 10 / 100

```

```

SE (tipo = 3)
    ENTÃO acresc ← gasto * 15 / 100
total ← gasto + acresc
tot_geral ← tot_geral + total
SE (total >= 500) E (total <= 1000)
    ENTÃO qtd_cons ← qtd_cons + 1
ESCREVA gasto
ESCREVA acresc
ESCREVA total
LEIA qtd
FIM
ESCREVA tot_geral
ESCREVA qtd_cons
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX38_A.PAS e \EXERC\CAP4\PASCAL\EX38_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX38_B.PAS e \EXERC\CAP4\PASCAL\EX38_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX38_A.CPP e \EXERC\CAP4\C++\EX38_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX38_B.CPP e \EXERC\CAP4\C++\EX38_B.EXE

39. Faça um programa que apresente o menu de opções a seguir, permita ao usuário escolher a opção desejada, receba os dados necessários para executar a operação e mostre o resultado. *Verificar a possibilidade de opção inválida e não se preocupar com restrições do tipo salário inválido.*

Menu de opções:

1. Imposto
2. Novo salário
3. Classificação
4. Finalizar o programa

Digite a opção desejada

Na opção 1: receber o salário de um funcionário, calcular e mostrar o valor do imposto usando as regras a seguir.

| SALÁRIOS | % DO IMPOSTO |
|----------------------------|--------------|
| Menor que R\$ 500,00 | 5 |
| De R\$ 500,00 a R\$ 850,00 | 10 |
| Acima de R\$ 850,00 | 15 |

Na opção 2: receber o salário de um funcionário, calcular e mostrar o valor do novo salário usando as regras a seguir.

| SALÁRIOS | AUMENTO |
|--|------------|
| Maiores que R\$ 1.500,00 | R\$ 25,00 |
| De R\$ 750,00 (inclusive) a R\$ 1.500,00 (inclusive) | R\$ 50,00 |
| De R\$ 450,00 (inclusive) a R\$ 750,00 | R\$ 75,00 |
| Menores que R\$ 450,00 | R\$ 100,00 |

Na opção 3: receber o salário de um funcionário e mostrar sua classificação usando a tabela a seguir.

| SALÁRIOS | CLASSIFICAÇÃO |
|----------------------------|----------------|
| Até R\$ 700,00 (inclusive) | Mal remunerado |
| Maiores que R\$ 700,00 | Bem remunerado |

ALGORITMO**Solução:**

```

ALGORITMO
DECLARE op, sal, imp, aum, novo_sal NUMÉRICO
REPITA
  ESCREVA "1- Imposto"
  ESCREVA "2- Novo Salário"
  ESCREVA "3- Classificação"
  ESCREVA "4- Finalizar o programa"
  ESCREVA "Digite a opção desejada"
  LEIA op
  SE ((op > 4) OU (op < 1))
    ENTÃO ESCREVA "Opção inválida !"
  SENÃO SE (op = 1)
    ENTÃO INÍCIO
      LEIA sal
      SE (sal < 500)
        ENTÃO imp ← sal * 5 / 100
      SE (sal >= 500) E (sal <= 850)
        ENTÃO imp ← sal * 10 / 100
      SE (sal > 850)
        ENTÃO imp ← sal * 15 / 100
      ESCREVA imp
    FIM
  SE (op = 2)
    ENTÃO INÍCIO
      LEIA sal
      SE (sal > 1500)
        ENTÃO aum ← 25
      SE (sal <= 750) E (sal <= 1500)
        ENTÃO aum ← 50
      SE (sal >= 450) E (sal < 750)
        ENTÃO aum ← 75
      SE (sal < 450)
        ENTÃO aum ← 100
      novo_sal ← sal + aum
      ESCREVA novo_sal
    FIM
  SE (op = 3)
    ENTÃO INÍCIO
      LEIA sal
      SE (sal <= 700)
        ENTÃO ESCREVA "Mal Remunerado"
      SENÃO ESCREVA "Bem Remunerado"
    FIM
  ATÉ op = 4
FIM_ALGORITMO.

```



RESOLUÇÃO
PASCAL

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX39_A.PAS e \EXERC\CAP4\PASCAL\EX39_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX39_B.PAS e \EXERC\CAP4\PASCAL\EX39_B.EXE



RESOLUÇÃO
C/C++

1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX39_A.CPP e \EXERC\CAP4\C++\EX39_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX39_B.CPP e \EXERC\CAP4\C++\EX39_B.EXE

40. Faça um programa que receba os dados a seguir de vários produtos: preço unitário, país de origem (1 – EUA, 2 – México e 3 – outros), meio de transporte (T – Terrestre, F – Fluvial e A – Aéreo), carga perigosa (S – Sim, N – Não). Calcule e mostre:

- ♦ o valor do imposto calculado usando a tabela a seguir.

| PREÇO UNITÁRIO | PERCENTUAL DE IMPOSTO SOBRE O PREÇO UNITÁRIO |
|----------------------|---|
| Até R\$ 100,00 | 5% |
| Maior que R\$ 100,00 | 10% |

- ♦ o valor de transporte calculado usando a tabela a seguir.

| CARGA PERIGOSA | PAÍS DE ORIGEM | VALOR DO TRANSPORTE |
|----------------|----------------|---------------------|
| S | 1 | R\$ 50,00 |
| | 2 | R\$ 35,00 |
| | 3 | R\$ 24,00 |
| N | 1 | R\$ 12,00 |
| | 2 | R\$ 35,00 |
| | 3 | R\$ 60,00 |

- ♦ o valor do seguro, calculado usando a regra a seguir.

Os produtos que vêm do México e os produtos que utilizam transporte aéreo pagam metade do valor do seu preço unitário como seguro.

- ♦ o preço final;
- ♦ o total dos impostos.



ALGORITMO

SOLUÇÃO:

```
ALGORITMO
DECLARE preco, imp, transp, segura, final NUMÉRICO
total_imp, origem NUMÉRICO
Meio_t, carga LITERAL
```



```

LEIA preco
ENQUANTO (preco > 0) FAÇA
INÍCIO
    LEIA origem
    LEIA meio_t
    LEIA carga
    SE (preco <= 100)
        ENTÃO imp ← preco * 5 / 100
        SENÃO imp ← preco * 10 / 100
    SE (carga = "S")
        ENTÃO INÍCIO
            SE (origem = 1)
                ENTÃO transp ← 50
            SE (origem = 2)
                ENTÃO transp ← 35
            SE (origem = 3)
                ENTÃO transp ← 24
        FIM
    SE (carga = "N")
        ENTÃO INÍCIO
            SE (origem = 1)
                ENTÃO transp ← 12
            SE (origem = 2)
                ENTÃO transp ← 35
            SE (origem = 3)
                ENTÃO transp ← 60
        FIM
    SE (origem = 2) OU (meio_t = "A")
        ENTÃO seguro ← preco/2
    SENÃO seguro ← 0
    final ← preco + imp + transp + seguro
    total_imp ← total_imp + imp
    ESCRIVA imp
    ESCRIVA transp
    ESCRIVA seguro
    ESCRIVA final
    LEIA preco
FIM
ESCREVA total_imp
FIM_ALGORITMO.

```



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\PASCAL\EX40_A.PAS e \EXERC\CAP4\PASCAL\EX40_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA REPEAT:

\EXERC\CAP4\PASCAL\EX40_B.PAS e \EXERC\CAP4\PASCAL\EX40_B.EXE



1ª SOLUÇÃO - UTILIZANDO A ESTRUTURA WHILE:

\EXERC\CAP4\C++\EX40_A.CPP e \EXERC\CAP4\C++\EX40_A.EXE

2ª SOLUÇÃO - UTILIZANDO A ESTRUTURA DO-WHILE:

\EXERC\CAP4\C++\EX40_B.CPP e \EXERC\CAP4\C++\EX40_B.EXE

EXERCÍCIOS PROPOSTOS

1. Faça um programa que verifique e mostre os números entre 1.000 e 2.000 (inclusive) que, quando divididos por 11, produzam resto igual a 5.

2. Faça um programa que leia um valor n , inteiro e positivo, calcule e mostre a seguinte soma:

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n}$$

3. Faça um programa que calcule e mostre o produto dos números primos entre 92 e 1.478.

4. Faça um programa que leia cinco grupos de quatro valores (A, B, C, D) e mostre-os na ordem lida. Em seguida, mostre-os em ordem crescente e decrescente.

5. Uma loja tem 15 clientes cadastrados e deseja enviar uma correspondência a cada um deles anunciando um bônus especial. Faça um programa que leia o nome do cliente e o valor de suas compras no ano passado. Calcule e mostre um bônus de 10% se o valor das compras for menor que R\$ 1.000,00 e de 15%, caso contrário.

6. Uma companhia de teatro deseja dar uma série de espetáculos. A direção calcula que a R\$ 5,00 o ingresso, serão vendidos 120 ingressos, e que as despesas serão R\$ 200,00. Diminuindo-se R\$ 0,50 o preço dos ingressos espera-se que as vendas aumentem em 26 ingressos.

Faça um programa que escreva uma tabela de valores de lucros esperados em função do preço do ingresso, fazendo-se variar esse preço de R\$ 5,00 a R\$ 1,00 de R\$ 0,50 em R\$ 0,50. Escreva, ainda, o lucro máximo esperado, o preço do ingresso e a quantidade de ingressos vendidos para a obtenção desse lucro.

7. Faça um programa que receba a idade de dez pessoas e que calcule e mostre a quantidade de pessoas com idade maior ou igual a 18 anos.

8. Faça um programa que receba a idade de 15 pessoas e que calcule e mostre:

- ◆ a quantidade de pessoas em cada faixa etária;
- ◆ a percentagem de pessoas na primeira e na última faixa etária, com relação ao total de pessoas.

| FAIXA ETÁRIA | IDADE |
|--------------|------------------|
| 1ª | Até 15 anos |
| 2ª | De 16 a 30 anos |
| 3ª | De 31 a 45 anos |
| 4ª | De 46 a 60 anos |
| 5ª | Acima de 61 anos |

9. Faça um programa que receba um número e que calcule e mostre a tabuada desse número.

10. Faça um programa que mostre as tabuadas dos números de 1 a 10.

11. Uma loja utiliza o código V para transação à vista e P para transação a prazo. Faça um programa que receba o código e o valor de 15 transações. Calcule e mostre:

- ◆ o valor total das compras à vista;
- ◆ o valor total das compras a prazo;
- ◆ o valor total das compras efetuadas;
- ◆ o valor da primeira prestação das compras a prazo, sabendo-se que essas serão pagas em três vezes.

12. Faça um programa que receba a idade, a altura e o peso de 25 pessoas. Calcule e mostre:

- ◆ a quantidade de pessoas com idade superior a 50 anos;
- ◆ a média das alturas das pessoas com idade entre 10 e 20 anos;
- ◆ a percentagem de pessoas com peso inferior a 40 quilos entre todas as pessoas analisadas.

13. Faça um programa que receba a idade e o peso de sete pessoas. Calcule e mostre:

- ◆ a quantidade de pessoas com mais de 90 quilos;
- ◆ a média das idades das sete pessoas.

14. Faça um programa que receba a idade, o peso, a altura, a cor dos olhos (A – Azul, P – Preto, V – Verde e C – Castanho) e a cor dos cabelos (P – Preto, C – Castanho, L – Louro e R – Ruivo) de 20 pessoas e que calcule e mostre:

- ◆ a quantidade de pessoas com idade superior a 50 anos e peso inferior a 60 quilos;
- ◆ a média das idades das pessoas com altura inferior a 1,50;
- ◆ a percentagem de pessoas com olhos azuis entre todas as pessoas analisadas;
- ◆ a quantidade de pessoas ruivas e que não possuem olhos azuis.

15. Faça um programa que receba dez números e que calcule e mostre a quantidade de números entre 30 e 90.

16. Faça um programa que receba dez idades, pesos e alturas e que calcule e mostre:

- ◆ a média das idades das dez pessoas;
- ◆ a quantidade de pessoas com peso superior a 90 quilos e altura inferior a 1,50;
- ◆ a percentagem de pessoas com idade entre 10 e 30 anos entre as pessoas que medem mais de 1,90.

17. Faça um programa que receba a idade e o sexo de sete pessoas e que calcule e mostre:

- ◆ a idade média do grupo;
- ◆ a idade média das mulheres;
- ◆ a idade média dos homens.

18. Faça um programa que receba dez números, calcule e mostre a soma dos números pares e a soma dos números primos.

19. Faça um programa que receba o valor de um carro e mostre uma tabela com os seguintes dados: preço final, quantidade de parcelas e valor da parcela. Considere o seguinte:

1. O preço final para compra à vista tem um desconto de 20%.
2. A quantidade de parcelas pode ser: 6, 12, 18, 24, 30, 36, 42, 48, 54 e 60.
3. Os percentuais de acréscimo seguem a tabela a seguir.

| QUANTIDADE DE PARCELAS | PERCENTUAL DE ACRÉSCIMO SOBRE O PREÇO FINAL |
|---------------------------|--|
| 6 | 3% |
| 12 | 6% |
| 18 | 9% |
| 24 | 12% |
| 30 | 15% |
| 36 | 18% |
| 42 | 21% |
| 48 | 24% |
| 54 | 27% |
| 60 | 30% |

20. Faça um programa que receba dez números inteiros e mostre a quantidade de números primos dentre os números que foram digitados.

21. Faça um programa para calcular $n!$ (fatorial de n), sendo que o valor inteiro de n é fornecido pelo usuário.

Sabe-se que:

$$n! = 1 * 2 * 3 * \dots * (n-1) * n;$$

$$0! = 1, \text{ por definição.}$$

22. Faça um programa que receba a idade e o peso de 15 pessoas. Calcule e mostre as médias dos pesos das pessoas da mesma faixa etária. As faixas etárias são: de 1 a 10 anos, de 11 a 20 anos, de 21 a 30 anos e maiores de 31 anos.

23. Cada espectador de um cinema respondeu a um questionário no qual constava sua idade e a sua opinião em relação ao filme: ótimo – 3, bom – 2, regular – 1. Faça um programa que receba a idade e a opinião de 15 espectadores e que calcule e mostre:

- ♦ a média das idades das pessoas que responderam ótimo;
- ♦ a quantidade de pessoas que respondeu regular;
- ♦ a percentagem de pessoas que respondeu bom entre todos os espectadores analisados.

24. Uma firma fez uma pesquisa de mercado para saber se as pessoas gostaram ou não de um novo produto lançado no mercado. Para isso forneceu o sexo do entrevistado e sua resposta (S – Sim ou N – Não). Sabe-se que foram entrevistadas dez pessoas. Faça um programa que calcule e mostre:

- ♦ o número de pessoas que respondeu sim;
- ♦ o número de pessoas que respondeu não;
- ♦ o número de mulheres que respondeu sim;
- ♦ a percentagem de homens que respondeu não entre todos os homens analisados.

25. O sistema de avaliação de uma determinada disciplina obedece aos seguintes critérios:

- ♦ durante o semestre são dadas três notas;
- ♦ a nota final é obtida pela média aritmética das três notas;
- ♦ é considerado aprovado o aluno que obtiver a nota final superior ou igual a 6 e que tiver comparecido a um mínimo de 40 aulas.

Faça um programa que:

- ◆ leia um conjunto de dados contendo o número da matrícula, as três notas e a frequência (número de aulas frequentadas) de dez alunos.

Calcule e mostre:

- ◆ para cada aluno o número da matrícula, a nota final e a mensagem (aprovado ou reprovado);
- ◆ a maior e a menor nota da turma;
- ◆ o total de alunos reprovados;
- ◆ a percentagem de alunos reprovados por frequência abaixo da mínima necessária.

26. Faça um programa que receba várias idades e que calcule e mostre a média das idades digitadas. Finalize digitando idade igual a zero.

27. Foi feita uma pesquisa de audiência de canal de TV em várias casas de uma cidade, em um determinado dia. Para cada casa consultada foi fornecido o número do canal (4, 5, 7, 12) e o número de pessoas que estavam assistindo àquele canal. Se a televisão estivesse desligada, nada era anotado, ou seja, essa casa não entrava na pesquisa. Faça um programa que:

- leia um número indeterminado de dados (número do canal e o número de pessoas que estavam assistindo);
- calcule e mostre a percentagem de audiência de cada canal.

Para encerrar a entrada de dados digite o número do canal ZERO.

28. A prefeitura de uma cidade fez uma pesquisa entre seus habitantes, coletando dados sobre o salário e o número de filhos. A prefeitura deseja saber:

- a média do salário da população;
- a média do número de filhos;
- o maior salário;
- a percentagem de pessoas com salários até R\$ 150,00.

O final da leitura de dados dar-se-á com a entrada de um salário negativo.

29. Foi feita uma pesquisa entre os habitantes de uma região. Foram coletados os dados de idade, sexo (M/F) e salário. Faça um programa que calcule e mostre:

- a média dos salários do grupo;
- a maior e a menor idade do grupo;
- a quantidade de mulheres com salário até R\$ 200,00;
- a idade e o sexo da pessoa que possui o menor salário.

Finalize a entrada de dados ao ser digitada uma idade negativa.

30. Uma empresa deseja aumentar seus preços em 20%. Faça um programa que leia o código e o preço de custo de cada produto e que calcule o novo preço. Calcule também a média dos preços com e sem aumento. Mostre o código e o novo preço de cada produto e, no final, as médias. A entrada de dados deve terminar quando for lido um código de produto negativo.

31. Faça um programa que receba o tipo da ação, ou seja, uma letra a ser comercializada na bolsa de valores, o preço de compra e o preço de venda de cada ação e que calcule e mostre:

- ◆ o lucro de cada ação comercializada;
- ◆ a quantidade de ações com lucro superior a R\$ 1.000,00;

- ♦ a quantidade de ações com lucro inferior a R\$ 200,00;
- ♦ o lucro total da empresa.

Finalize com o tipo de ação 'F'.

32. Faça um programa que receba vários números e que calcule e mostre:

- ♦ a quantidade de números inferiores a 35;
- ♦ a média dos números positivos;
- ♦ a percentagem de números entre 50 e 100 entre todos os números digitados;
- ♦ a percentagem de números entre 10 e 20 entre os números menores que 50.

33. Faça um programa que apresente o menu de opções a seguir:

Menu de opções:

1. Média aritmética
2. Média ponderada
3. Sair

Digite a opção desejada

Na opção 1: receber duas notas, calcular e mostrar a média aritmética.

Na opção 2: receber três notas e seus respectivos pesos, calcular e mostrar a média ponderada.

Na opção 3: sair do programa.

Verifique a possibilidade de opção inválida, mostrando uma mensagem.

34. Em uma eleição presidencial existem quatro candidatos. Os votos são informados por meio de código. Os códigos utilizados são:

| 1, 2, 3, 4 | VOTOS PARA OS RESPECTIVOS CANDIDATOS |
|------------|--------------------------------------|
| 5 | Voto nulo |
| 6 | Voto em branco |

Faça um programa que calcule e mostre:

- a) o total de votos para cada candidato;
- b) o total de votos nulos;
- c) o total de votos em branco;
- d) a percentagem de votos nulos sobre o total de votos;
- e) a percentagem de votos em branco sobre o total de votos.

Para finalizar o conjunto de votos, tem-se o valor zero.

35. Faça um programa que receba como entrada uma lista de números positivos ou negativos, terminada com o número zero. O programa deve fornecer como saída a soma dos números positivos, a soma dos números negativos e a soma das duas somas parciais.

36. Faça um programa que receba a idade e a altura de várias pessoas e que calcule e mostre a média das alturas das pessoas com mais de 50 anos. Para encerrar a entrada de dados digite idade menor ou igual a zero.

37. Faça um programa que apresente um menu de opções para o cálculo das seguintes operações entre dois números: adição, subtração, multiplicação e divisão. O programa deve possibilitar ao usuário a escolha da operação desejada, a exibição do resultado e a volta ao menu de opções. O programa só termina quando for escolhida a opção de saída.

38. Faça um programa que apresente o menu de opções a seguir, que permita ao usuário escolher a opção desejada, receba os dados necessários para executar a operação e mostre o resultado. Verificar a possibilidade de opção inválida e não se preocupar com as restrições, como salário inválido.

Menu de opções:

1. Novo salário
2. Férias
3. Décimo terceiro
4. Sair

Digite a opção desejada

Na opção 1: receber o salário de um funcionário, calcular e mostrar o novo salário usando as regras a seguir.

| SALÁRIOS | PERCENTAGEM DE AUMENTO |
|----------------------------|------------------------|
| Até R\$ 350,00 | 15% |
| De R\$ 350,00 a R\$ 600,00 | 10% |
| Acima de R\$ 600,00 | 5% |

Na opção 2: receber o salário de um funcionário, calcular e mostrar o valor de suas férias. Sabe-se que as férias equivalem ao seu salário acrescido de $\frac{1}{3}$.

Na opção 3: receber o salário de um funcionário e o número de meses de trabalho na empresa, no máximo 12, calcular e mostrar o valor do décimo terceiro. Sabe-se que o décimo terceiro equivale ao seu salário multiplicado pelo número de meses de trabalho dividido por 12.

Na opção 4: sair do programa.

39. Faça um programa que receba um conjunto de valores inteiros e positivos e que calcule e mostre o maior e o menor valor do conjunto. Considere que:

- ♦ para encerrar a entrada de dados, deve ser digitado o valor zero;
- ♦ para valores negativos, deve ser enviada uma mensagem;
- ♦ os valores negativos ou iguais a zero não entrarão nos cálculos.

40. Uma agência bancária possui vários clientes que podem fazer investimentos com rendimentos mensais, conforme a tabela a seguir:

| TIPO | DESCRIÇÃO | RENDIMENTO MENSAL |
|------|----------------------|-------------------|
| 1 | Poupança | 1,5 % |
| 2 | Poupança plus | 2 % |
| 3 | Fundos de renda fixa | 4 % |

Faça um programa que leia o código do cliente, o tipo da conta e o valor investido e que calcule e mostre o rendimento mensal de acordo com o tipo do investimento. Ao final do programa mostre o total investido e o total de juros pagos.

A leitura terminará quando o código do cliente digitado for menor ou igual a 0.