

1º quesito (5 pontos): Escolha e resolva uma das opções abaixo:

1.a – Escreva um programa em PASCAL que crie um pequeno cadastro de imóveis para aluguel de uma imobiliária (Número máximo de imóveis igual a 20). Armazene os dados dos imóveis num vetor de registro com a seguinte estrutura:

- endereço (string com 30 posições)
- bairro (string com 20 posições).
- nome do proprietário (string com 20 posições)
- valor do aluguel (real)
- número de quartos (integer)
- salão de festas (boolean: TRUE → tem salão de festa; FALSE → não tem salão)
- nº de vagas para carro (integer)

Após a criação do cadastro, o programa deve, em processo repetitivo, solicitar o nome de um bairro, o número de vagas para carro e o valor máximo do aluguel desejado e, em seguida, apresentar na tela a relação de todos os imóveis cadastrados para o bairro solicitado e que tenham o nº de vagas solicitado e com valor do aluguel ≤ ao aluguel desejado. O processo repetitivo termina quando for informado para o bairro a palavra FIM.

Obs.: 1) O número de imóveis a serem cadastrados deve ser lido no início do programa;

2) O programa deve conter necessariamente:

- a) Um procedimento que receba o número de imóveis a serem cadastrados e retorne por parâmetro o vetor registro com os imóveis cadastrados;
- b) Um procedimento para imprimir a relação dos imóveis para o bairro solicitado e que tenham as características indicadas.

Resposta:

```
PROGRAM PRIMEIRO_A;
USES CRT;
TYPE REG = RECORD
    ENDER: STRING[30];
    BAIRRO, NOME: STRING[20];
    ALUG: REAL;
    NQRT: INTEGER;
    SALAO: BOOLEAN;
    NVC: INTEGER;
END;
VET = ARRAY[1..20] OF REG;
VAR NI: INTEGER; {Número de imóveis a serem cadastrados}
    CAD: VET; {Vetor de registros com os campos referentes aos dados dos imóveis}
    BAIRRO: STRING[20]; {Nome do bairro para seleção dos imóveis a serem listados}
    VAGAS: INTEGER; {Número de vagas para carros}
    ALUGUEL: REAL; {Valor do aluguel informado}
    SN: CHAR; {Resposta: Sim/ Não}
PROCEDURE CADAstra(N: INTEGER; VAR CD: VET);
VAR I: INTEGER;
BEGIN
    FOR I:= 1 TO N DO
        BEGIN
            WRITE('Endereço: '); READLN(CD[I].ENDER);
            WRITE('Bairro: '); READLN(CD[I].BAIRRO);
            WRITE('Áluguel: '); READLN(CD[I].ALUG);
            WRITE('N. de quartos: '); READLN(CD[I].NQRT);
            WRITE('Salão de festa? ("S"/"N") '); READLN( SN );
            IF SN = 'S' THEN CD[I].SALAO:= TRUE ELSE CD[I].SALAO:= FALSE;
            WRITE('N. de vagas: '); READLN(CD[I].NVC);
        END;
    END;
PROCEDURE LISTA(N: INTEGER; CD: VET; BR: STRING[20]; NV:INTEGER; ALUG: REAL);
VAR I: INTEGER;
BEGIN
    FOR I:= 1 TO N DO
        BEGIN
            IF (CD[I].BAIRRO = BR) AND (CD[I].NVC = NV) AND (CD[I].ALUG <= ALUG) THEN
                BEGIN
                    WRITELN( 'Endereço: ', CD[I].ENDER);
                    WRITELN( 'Bairro: ', CD[I].BAIRRO);
                    WRITELN( 'Aluguel: ', CD[I].ALUG:7:2);
                    WRITELN( ' N. de quartos: ', CD[I].NQRT);
```

```

        WRITE( ' Salão de festas? ' );
        IF CAD[I].SALAO THEN WRITELN( ' SIM' ) ELSE WRITELN( ' NÃO' );
        WRITELN( ' N. de vagas: ', CD[I].NVC);
        WRITELN( ' -----'); {Para separar os imóveis}
    END;
END;
END;
BEGIN {Begin principal do programa}
    READLN( NI ); {Leitura do número de imóveis a serem cadastrados}
    IF (NI > 0) AND (NI <= 20) THEN
    BEGIN {2}
        CADASTRA(NI, CAD); {Cria cadastro de imóveis}
        CLRSCR;
        WRITE( 'Informe o bairro ("FIM" para terminar) ); READLN( BAIRRO );
        WHILE BAIRRO <> 'FIM' DO
        BEGIN {3}
            WRITE( 'N. de vagas ? '); READ( VAGAS );
            WRITE( 'Valor máximo do aluguel ?' ); READ( ALUGUEL );
            LISTA( NI, CAD, BAIRRO, VAGAS, ALUGUEL); {Lista os imóveis com as características solicitadas}
            READKEY;
            WRITE( 'Informe o bairro ("FIM" para terminar) ); READLN( BAIRRO );
        END; {3}
    END {2} ELSE WRITE( 'Número de imóveis a serem cadastrados fora de limites' );
END. {Final do programa}

```

1.b – Fazer um programa pascal para:

- Ler um valor para N, inteiro, maior que 2 e menor ou igual a 10;
- Ler, a partir de teclado, dois vetores A e B de elementos inteiros, com N elementos cada um, utilizando o procedimento descrito na obs. 1;
- Construir o vetor X formado pelos elementos de A que estão contidos em B, utilizando a função descrita na obs. 2.
Sugestão: percorrer todos os elementos do vetor A e, para cada um desses elementos, utilizar a função proposta para verificar se o valor do elemento em questão está contido em B, gravando-o ou não no vetor X, conforme o resultado da função seja TRUE ou FALSE..
- Imprimir os vetores A, B, utilizando o procedimento descrito na obs. 3;
- Imprimir o vetor X, utilizando o mesmo procedimento descrito na obs. 3, ou a mensagem “Os vetores A e B não têm elementos em comum” conforme o caso.

Obs. Incluir no programa

- um procedimento para ler um vetor (e somente um) de elementos inteiros;
- uma função que receba por parâmetros um valor inteiro K e um vetor W, retornando com o valor TRUE, se o valor de K está contido em W ou o valor FALSE, em caso contrário;
- um procedimento para imprimir um vetor (e somente um) de elementos inteiros;

Resposta:

```

PROGRAM PRIMEIRO_B;
CONST NMAX = 10;
TYPE VET = ARRAY[1..NMAX] OF INTEGER;
VAR N, I, NX: INTEGER;
    A, B, X: VET;
PROCEDURE LERVETOR(K: INTEGER; VAR V: VET);
    VAR I: INTEGER;
BEGIN
    FOR I:= 1 TO K DO
        READ( V[I] );
    END;
FUNCTION CONTIDO( VAL, K: INTEGER; V: VET): BOOLEAN;
    VAR I: INTEGER;
BEGIN
    CONTIDO:= FALSE;
    FOR I:= 1 TO K DO
        IF VAL = V[I] THEN CONTIDO:= TRUE;
    END;
PROCEDURE IMPVETOR(K: INTEGER; V: VET);
    VAR I: INTEGER;
BEGIN
    FOR I:= 1 TO K DO
        WRITE( V[I]:4)
    WRITELN
END;

```

```

BEGIN
  REPEAT
    WRITE( 'N = ? ' ); READLN( N );
  UNTIL ( N > 2 ) AND ( N <= NMAX );
  WRITELN( 'Digite o vetor A' ); LERVETOR( N, A );
  WRITELN( 'Digite o vetor B' ); LERVETOR( N, B );
  NX:= 0;
  FOR I:= 1 TO N DO
    IF CONTIDO( A[I], N, B ) THEN BEGIN
      NX:= NX + 1;
      X[NX]:= A[I];
    END;
  WRITELN; WRITELN( ' Vetor A' ); IMPVETOR( N, A );
  WRITELN; WRITELN( ' Vetor B' ); IMPVETOR( N, B ); WRITELN;
  IF NX = 0 THEN WRITE( 'Os vetores A e B não têm elementos em comum' ) ELSE
  BEGIN WRITELN( ' Vetor X' ); IMPVETOR( NX, X ) END;
END.

```

2º quesito (5 pontos)

O Comitê Olímpico dispõe de um arquivo, denominado “MEDALHAS.BRA” com dados dos atletas e o número de medalhas obtidas nos últimos jogos olímpicos, com o seguinte modelo de registro:

Número	Nome	Endereço	Sigla do estado	Medalhas de ouro	Medalhas de prata	Medalhas de bronze
Integer	String[30]	String[50]	String[2]	Integer	Integer	Integer

Pede-se: fazer um programa em PASCAL para criar um outro arquivo denominado “MEDALHAS.EST”, à partir do já existente, com os atletas de um determinado estado e com a estrutura de registro apresentada ao lado:

Nome	N. total de medalhas
String[30]	Integer

O programa deve inicialmente ler a sigla do estado e em seguida criar o novo arquivo com os atletas do estado e que obtiveram pelo menos uma medalha de ouro nos jogos olímpicos.

Resposta:

```

PROGRAM SEGUNDO;
  TYPE REG1 = RECORD
    NUM: INTEGER;
    NOME: STRING[30];
    ENDER: STRING[50];
    SIGLA: STRING[2];
    NMO, NMP, NMB: INTEGER;
  END;
  TYPE REG2 = RECORD
    NOME: STRING[30];
    NTM: INTEGER;
  END;
  VAR ESTADO: STRING[2];
      ATLBRA: REG1;
      ARQBRA: FILE OF REG1;
      ATLEST: REG2;
      ARQUEST: FILE OF REG2;
BEGIN
  WRITE( 'Informe o estado desejado' ); READLN( ESTADO );
  ASSIGN( ARQBRA, 'MEDALHAS.BRA' ); RESET( ARQBRA );
  ASSIGN( ARQUEST, 'MEDALHAS.EST' ); REWRITE( ARQUEST );
  WHILE NOT EOF( ARQBRA ) DO
  BEGIN
    READ( ARQBRA, ATLBRA );
    IF ( ATLBRA.SIGLA = ESTADO ) AND ( ATLBRA.NMO > 0 ) THEN
    BEGIN
      ATLEST.NOME:= ATLBRA.NOME;
      ATLEST.NTM:= ATLBRA.NMO + ATLBRA.NMP + ATLBRA.NMB;
      WRITE( ARQUEST, ATLEST );
    END;
  END;
  CLOSE( ARQBRA ); CLOSE( ARQUEST );
END.

```

Feliz Natal