Nota: 3,0

Considerações sobre a Avaliação.

1. Todos receberão uma prova composta de 4 questões valendo um total de 10 pontos.
2. A prova terá uma duração máxima de 120 minutos.
3. A organização da prova e dos códigos, incluindo indentação, fazem parte da avaliação.
4. A prova deverá ser realizada com caneta de tinta azul ou preta.
5. Todos os códigos devem ser escritos em Linguagem C.
6. Outras observações, ou modificações das citadas acima, podem ser feitas durante a prova pelo professor.

Em todas as questões a seguir, utilize apenas aritmética de ponteiros para manipular as matrizes/vetores.

Questão 1: [2½ pontos]

Considere os vetores x e y, ambos de tamanho n, alocados dinamicamente. Dado que a distância d entre dois pontos correspondentes (de mesma posição) em x e y é dado por:

$$d_i = |x_i - y_i|$$

Organize em v os endereços de x e y com menor d.

Questão 2: [2½ pontos]

Dado um vetor x de inteiros e de tamanho n. Dado um valor t também inteiro. Escreva uma função que modifique x conforme a regra a seguir:

$$x_i = \begin{cases} 1, & \text{se } x_i > t \\ 0, & \text{caso contrário} \end{cases}$$

Necessário desenvolver a função main().

Questão 3: [2½ pontos]

Um vetor $x : array[1..n]$ de inteiros é dado tal que $x[1] \leq x[2] \leq \ldots \leq x[n]$. Escreva uma função que retorne um valor z conforme a regra a seguir:

$$z = \begin{cases} x[(n+1)/2], & \text{se } n \text{ for ímpar} \\ \frac{x[n/2]+x[(n/2)+1]}{2}, & \text{caso contrário} \end{cases}$$

NÃO é necessário desenvolver a função main().

Questão 4: [2½ pontos]

Dado uma string s e um caractere c, implemente uma função que escreva a quantidade de ocorrências de c em s em uma variável recebida por referência.

NÃO é necessário desenvolver a função main().

#INCLUDE <STDIO.H>
#INCLUDE <STDLIB.H>

P_VET1 == V[0]

```
INT MAIN() {
    INT *P_VET1, *P_VET2, N, *P_D, MENOR;
    PRINTF("DIGITE O TAMANHO DOS VETORES 1 E 2:\n");
    SCANF("%D", &N);
    P_VET1 = MALLOC(N * SIZEOF(INT));
    P_VET2 = MALLOC(N * SIZEOF(INT));

    FOR(INT I=0; I<N; I++){
        P_D = *(P_VET1 + I) - *(P_VET2 + I);
        MENOR = *P_D;
        IF (*P_D) < MENOR){
            MENOR = *P_D;
        }
    }
```

Qual o valor inicial?

7

0,5

-> NOME: ANA LETÍCIA MACIEL COSTA

02 ->
```
#INCLUDE <STDIO.H>
# INCLUDE <STDLIB.H>

#DEFINE N 10

VOID CHANGE_VET (INT, INT, INT);

INT MAIN() {
    INT VET_X[N], NUM_T, *P_VET_X, *P_NUM_T;
      P_VET_X = VET_X;
    FOR(INT I=0; I<N; I++ ){
        PRINTF("DIGITE O %dº NÚMERO DO VETOR X: \n", I+1);
        SCANF("%d", P_VET_X +I );
    }

    PRINTF("DIGITE UM NÚMERO: \n");
    SCANF("%d", &NUM_T);
    P_NUM_T = &NUM_T

    CHANGE_VET(*P_NUM_T, N, *P_VET_X);
    RETURN 0;
}

VOID CHANGE_VET(INT *P_NUM, INT TAM, INT * P_VET){
    FOR(INT I=0 ; I<TAM ; I++){
        IF(*(P_VET+I) > *P_NUM ){
            *(P_VET+I) = 1;
        }
        ELSE {
            *(P_VET+I) = 0;
        }
    }
}
```

2,5

```
#INCLUDE <STDIO.H>
#INCLUDE <STDLIB.H>

INT RETUR_VALUE(INT, INT, INT);

INT MAIN(){
   ...
}

INT RETURN_VALUE(INT Z, INT *P_VET, INT T){
   FOR(INT I=0; I<T; I++){
      IF((P_VET+I)%2==0){
         Z=((*(P_VET+I)/2)+(*(P_VET+I)/2)+J),
                              2
      }
      ELSE{
         Z=(*(P_VET+I)+J)
                  2;
      }
   RETURN Z;
}
```

-> NOME: ANA LETÍCIA MACIEL COSTA

04 ->

```
#INCLUDE <STDIO.H>
#INCLUDE <STDLIB.H>
#INCLUDE <STRING.H>

VOID   COUNT_LETTER(CHAR, CHAR, INT);

INT MAIN(){
   ...
}

VOID  COUNT_LETTER(CHAR *P_LET, CHAR *P_STR, INT *COUNT){
      WHILE(*(P_STR) != '\0'){
          IF(*(P_STR) == *P_LET){
           *COUNT++;
          }
          P_STR++;
      }

PRINTF("O NÚMERO DE VEZES QUE A LETRA %C APARECEU FOI DE %D\N", *P_LET, COUNT);

};
```

Tem que receber o endereço para poder ser acessível externamente.

2,0