

PROVA – ESTRUTURA DE DADOS

ATUALIZADA DIA 06/10

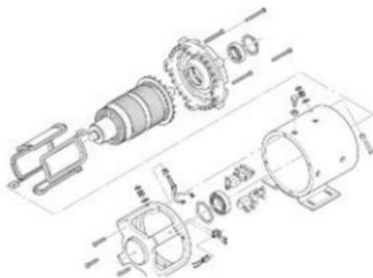
➤ QUESTÃO 1:

1. Leia o texto e responda à questão a seguir:

Roberto Fittichello, conhecido como Betão, leu o artigo:

Oficina mecânica novas atitudes

<http://www.administradores.com.br/artigos/tecnologia/oficina-mecanica-novas-attitudes/65518/> Novas tecnologias e o aumento da concorrência, fizeram o surgimento de novas exigências, o que torna necessário investir na modernização dos conceitos administrativo “Quando o assunto era oficinas mecânicas logo aparece na lembrança as pequenas, abafadas e pouco profissionais oficinas, cuja qualidade do atendimento dependia do humor de seu proprietário. Com a chegada de novas tecnologias e o aumento da concorrência, fizeram o surgimento de novas exigências, o que torna necessário investir na modernização dos conceitos administrativos e dos serviços para competir no ramo. Os serviços de uma oficina mecânica é constituído por proprietários particulares de automóveis e clientes corporativos, incluindo-se seguradoras, cooperativas de transporte, empresas ou pessoas que possuem diversos carros ou frota própria de veículo, tais como órgãos públicos, empresas de energia, telefonia etc. (...) Com a maior concorrência do setor, exige-se maior criatividade do futuro empreendedor para identificação do seu nicho de mercado considerando-se além de seus conhecimentos e orçamento de investimento. novas tecnologias e o aumento da concorrência, fizeram o surgimento de novas exigências, o que torna necessário investir na modernização dos conceitos administrativos. (...) Betão sentiu que ali estava a imagem do seu negócio e que se ele não modernizasse fatalmente a sua oficina iria fechar as portas. Comentando alguns problemas, ele mostrou a oficina para um sobrinho que estava estudando na Universidade. O sobrinho observou que Roberto sempre desmontava as partes mecânicas para trocar alguma peça quebrada, ele colocava cuidadosamente cada peça que ia retirando, camada por camada, da parte externa até a parte mais interna em uma bacia para não perder os componentes. Depois de examinar e fazer a troca da peça defeituosa ele voltava a montar da parte mais interna até a parte mais externa buscando cada peça na bacia. Muitas vezes, e não eram poucos, observou que o tio tinha dificuldades pois as peças se misturavam na bacia e nem sempre um parafuso não era o adequado para aquela operação. Na oficina também trocava óleo. As latas ficavam empilhadas ao lado do escritório de Roberto, a demanda era até razoável, mas sempre havia perda pois as muitas vezes quando tinha um maior volume de vendas mesmo tendo óleo em estoque ele estava vencido inadequada para o uso.” O sobrinho de Roberto, como estava estudando estrutura de dados na Universidade, logo viu a oportunidade de melhorar a oficina propondo:



I- Aplicar a pilha na montagem e desmontagem mecânica alinhando as peças e Pilha no estoque do óleo.

II- Aplicar a Pilha na montagem e desmontagem mecânica alinhando as peças e Fila no estoque do óleo.

III- Aplicar a Fila tanto na montagem e desmontagem mecânica alinhando as peças como no estoque do óleo.

Podemos afirmar que:

- a. Apenas a proposta I melhora o processo de trabalho da oficina do Betão.
- b. **Apenas a proposta II melhora o processo de trabalho da oficina do Betão.**
- c. Apenas a proposta III melhora o processo de trabalho da oficina do Betão.
- d. Mais de uma das propostas melhora o processo de trabalho da oficina do Betão.

e. Nenhuma das propostas melhora o processo de trabalho da oficina do Betão

➤ QUESTÃO 2:

Com relação as filas, listas encadeadas e pilhas podemos afirmar:

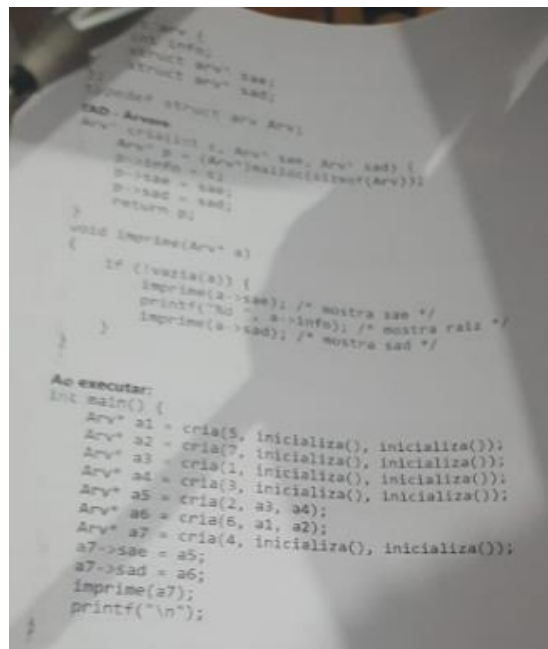
- a) Filas seguem o conceito LIFO, ou seja, o primeiro elemento que entra na estrutura é o primeiro que sai, listas encadeadas seguem o conceito FIFO das pilhas, ou seja, o primeiro que entra é o primeiro que sai.
- b) Pilhas seguem o conceito FIFO, ou seja, o último elemento que entra na estrutura é o último que sai, listas encadeadas seguem o conceito LIFO das filas, ou seja, o último que entra é o primeiro que sai.
- c) Filas seguem o conceito FIFO, ou seja, o primeiro elemento que entra na estrutura é o último que sai, lista encadeadas não seguem o conceito LIFO das pilhas, ou seja, o último que entra é o último que sai.
- d) Filas seguem o conceito LIFO, ou seja, o último elemento que entra na estrutura é o último que sai encadeadas não seguem o conceito FIFO das pilhas, ou seja, o primeiro que entra é o primeiro que sai.
- e) Filas seguem o conceito FIFO, ou seja, o primeiro elemento que entra na estrutura é o primeiro que sai, pilhas não seguem o conceito FIFO, na pilha o primeiro que entra é o último que sai.

➤ QUESTÃO 3:

Considere o TAD de uma árvore:

Qual a saída após a execução?

- a) 5 7 1 3 2 6 4
- b) 1 2 3 4 5 6 7
- c) 3 2 1 4 7 5 6
- d) 7 6 5 4 3 2 1
- e) 4 2 1 3 6 5 7



➤ QUESTÃO 4:

Dada as seguintes afirmações em relação à árvore binária de busca:

- I. Uma das possibilidades da remoção de um nó com um filhos é substituindo-se a valor do nó

a ser retirado pela chave do nó mais à direita da subárvore esquerda.

- II. A remoção de uma folha com dois filhos: o filho de maior valor substitui o pai.
- III. A remoção na folha: o nó filho sobe para a posição do pai.

Assinale a alternativa correta em relação às afirmações anteriores:

- a) Apenas a afirmação I é correta
- b) Apenas a afirmação II é correta
- c) Apenas a afirmação III é correta
- d) Nenhuma afirmação está correta (ESTÁ CORTADO NA PROVA, MAS NENHUMA DAS AFIRMAÇÕES ESTÁ OK)
- e) Todas as afirmações estão corretas



QUESTÃO 5:

Considerando a estrutura a seguir:

```
typedef struct pessoa{ char nome[20];
```

```
int idade;
```

```
}Pessoa;
```

```
Typedef struct aluno{ Pessoa p;
```

```
char turma;
```

```
}Aluno
```

Qual das alternativas a seguir cria um registro de aluno com as seguintes informações:

Nome Edvio

Idade:22

Turma:A

A) Aluno.turma='A';
Aluno.Pessoa.idade = 22;
strcpy(Aluno.Pessoa.nome,"Edvio");

B) Aluno a;
Pessoa p; Turma = 'A'; Idade=22
Strpy(nome,"Edvio");

C) Aluno a;
a.turma='A'; a.p.idade=22;
strcpy(a.p.nome,"Edvio");

D) Aluno a;
Pessoa p;
a.turma='A';

```
a.p.idade=22;  
strcpy(p.a.nome,"Edvio");
```

```
E) Aluno a;  
Pessoa p; a.turma='A';  
p.idade=22;  
strcpy(a.p.nome,"Edvio");
```

➤ **QUESTÃO 6:**

Se a música Casa do Emicida fosse um programa computacional: Casa Emicida
Lá fora é selva

A sós entre luz e trevas
Nós, presos nessas fases de guerra, medo e monstros, tipo Jogos Vorazes É pau, é pedra, é míssil
E crer, é cada vez mais difícil Entende o negócio: nunca foi fácil Solo não dócil, esperança fóssil
O samba deu conselhos: ouça Jacaré que dorme, vira bolsa, amor Eu disse no começo
É quem tem valor versus quem tem preço Segue teu instinto, que ainda é
Deus e o Diabo na terra do sol Onde a felicidade, se pisca, é isca E a realidade, trisca, anzol
Corre!

O céu é meu pai A terra, mamãe
E o mundo inteiro é tipo a minha casa O céu é meu pai
A terra, mamãe
E o mundo inteiro é tipo a minha casa

Aos quinze, o Saara na ampulheta Aos trinta, tempo é treta
Rápido como um cometa
Hoje a fé numa gaiola, o sonho na gaveta Foi pelo riso delas que vim
No mesmo caminho por nós, tipo Mágico de Oz Meu coração é tamborim, tem voz, sim
Ainda bate veloz
Entre drones e almas, flores e sorte Se não me matou, me fez forte
É o caos como cais; sem norte
Venci de teimoso, zombando da morte Sem amor, uma casa é só moradia
De afeto, vazia Tijolo e teto, fria
Sobre chances, é bom vê-las
Às vezes se perde o telhado, pra ganhar as estrelas Tendeu?

O céu é meu pai A terra, mamãe
E o mundo inteiro é tipo a minha casa O céu é meu pai
A terra, mamãe
E o mundo inteiro é tipo a minha casa

Ah, a gente já se acostumou Que a alegria pode ser breve Mostre o sorriso, tenha juízo A inveja
tem sono leve
À espreita, pesadelos São como desfiladeiros Chão, em brasa
Nunca se esqueça o caminho de casa

O céu é meu pai A terra, mamãe

E o mundo inteiro é tipo a minha casa O céu é meu pai

A terra, mamãe

E o mundo inteiro é tipo a minha casa

- Assinale o trecho adequado para modularizar e formar um procedimento:

a.

Lá fora é selva

A só entre luz e trevas

Nós, presos nessas fases de guerra, medo e monstros, tipo Jogos Vorazes É pau, é pedra, é míssil

E crer, é cada vez mais difícil Entende o negócio: nunca foi fácil Solo não dócil, esperança fóssil

O samba deu conselhos: ouça Jacaré que dorme, vira bolsa, amor Eu disse no começo

É quem tem valor, versus quem tem preço Segue teu instinto, que ainda é

Deus e o Diabo na terra do sol

Onde a felicidade, se pisca, é isca e a realidade, trisca, anzol

Corre!

b.

O céu é meu pai A terra, mamãe

E o mundo inteiro é tipo a minha casa O céu é meu pai

A terra, mamãe

E o mundo inteiro é tipo a minha casa

c.

Aos quinze, o Saara na ampulheta Aos trinta, tempo é treta

Rápido como um cometa

Hoje a fé numa gaiola, o sonho na gaveta Foi pelo riso delas que vim

No mesmo caminho por nós, tipo Mágico de Oz Meu coração é tamborim, tem voz, sim

Ainda bate veloz

Entre drones e almas, flores e sorte Se não me matou, me fez forte

É o caos como cais; sem norte

Venci de teimoso, zombando da morte Sem amor, uma casa é só moradia

De afeto, vazia Tijolo e teto, fria

Sobre chances, é bom vê-las

Às vezes se perde o telhado, pra ganhar as estrelas Tendeu?

d.

Ah, a gente já se acostumou Que a alegria pode ser breve Mostre o sorriso, tenha juízo A

inveja tem sono leve

À espreita, pesadelos São como desfiladeiros Chão, em brasa

Nunca se esqueça o caminho de casa

e.

míssil, difícil, fácil, fóssil

QUESTÃO 7:

Faça a associação entre os nós de controle e os seus conceitos:

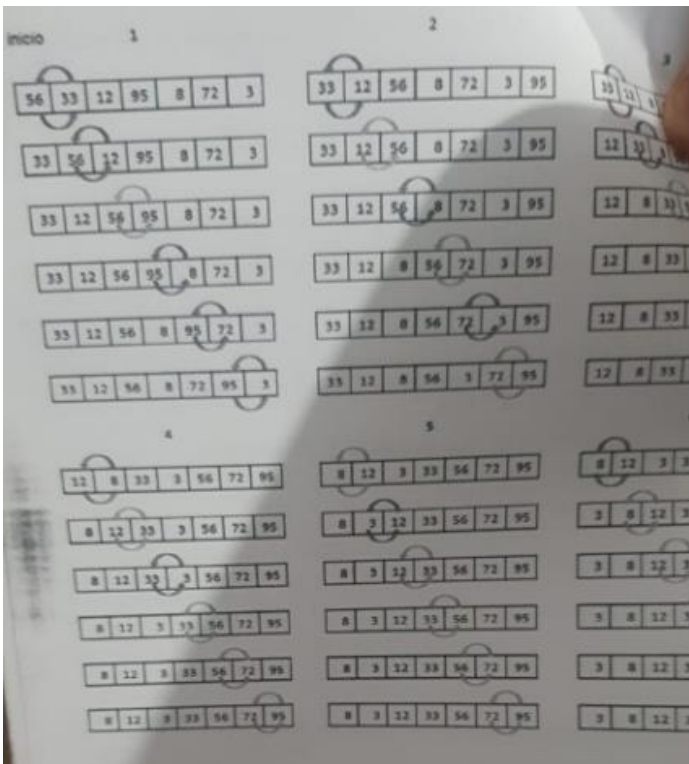
I	Fim	A	FIFO.
II	Topo	B	Lista Encadeada
III	Início fim	C	LIFO.

Assinale a correta:

- a) I-A; II-B; III-C
- b) I-B; II-C; III-A**
- c) I-A; II-C; III-B
- d) I-C; II-A; III-B
- e) I-B; II-A; III-C

QUESTÃO 8:

Um determinado algoritmo de ordenação de números inteiros em um vetor gerou (...) uma ordenação:



Assinale a alternativa com o método de ordenação empregada anteriormente:

- a) Quick Sort
- b) Merge Sort**

- c) Selection Sort
- d) Insertion Sort
- e) Bubble Sort



QUESTÃO 9:

Uma estrutura de dados tem o nó:

```
Typedef struct no {  
    Char info;  
    Struct no *próximo;  
}No;
```

Tem as funções Ins() para armazenar um valor na estrutura e Del() para retirar. Ao efetuar as operações:

```
Ins('f')  
Ins('k')  
Del()  
Ins('w')  
Ins('y')  
Del()
```

Resta na estrutura wy..

Assinale a alternativa com a estrutura restante após as seguintes operações iniciadas do valor

```
Ins('a')  
Ins('b')  
Ins('c')  
Del()  
Del()  
Ins('d')  
Ins('e')  
Del()  
Del()
```

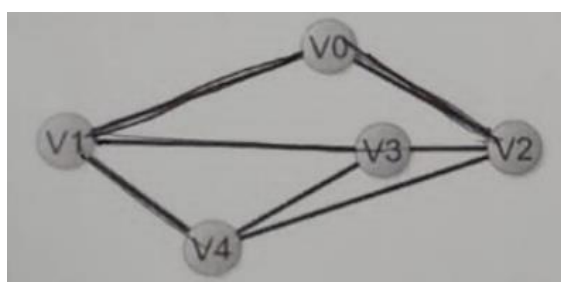
- a) Vazio
- b) c
- c) a
- d) e
- e) b



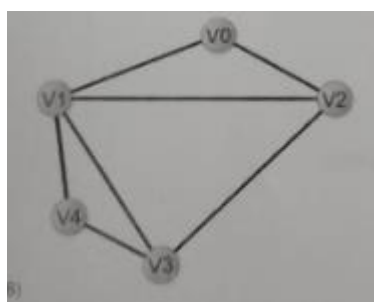
QUESTÃO 10:

Considere a matriz de adjacência a seguir:

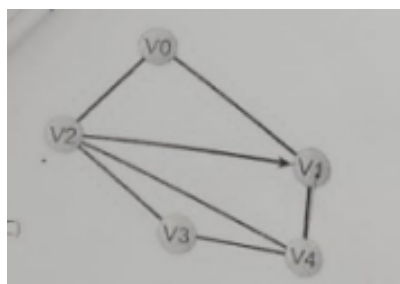
	V0	V1	V2	V3	V4
V0		1	1		
V1	1		1	1	1
V2	1			1	
V3		1	1		1
V4		1		1	



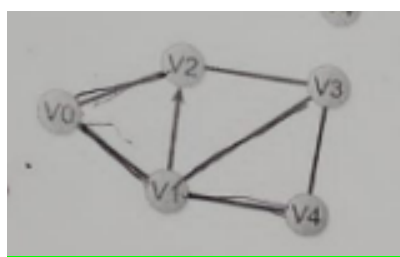
a)



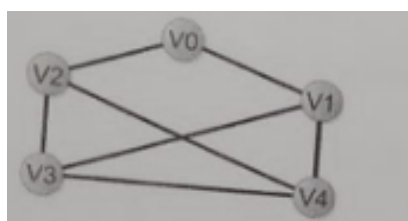
b)



c)



d)



e)

➤ **QUESTÃO 11:**

Considere o grafo criado a seguir:

Assinatura:

criaAresta(GRAFO* gr, int verticeInicial, int verticeFim, PESO p)

```
Int main(void) {  
    GRAFO* gr = criaGrafi(5);  
    criaAresta(gr, 0, 1, 2);  
    criaAresta(gr, 1, 0, 2);  
    criaAresta(gr, 0, 2, 7);  
    criaAresta(gr, 2, 0, 7);  
    criaAresta(gr, 1, 3, 3);  
    criaAresta(gr, 3, 1, 3);  
    criaAresta(gr, 1, 4, 1);  
    criaAresta(gr, 4, 1, 1);  
    criaAresta(gr, 2, 3, 6);  
    criaAresta(gr, 3, 2, 6);  
    criaAresta(gr, 2, 4, 3);  
    criaAresta(gr, 4, 2, 3);  
    criaAresta(gr, 3, 4, 4);  
    criaAresta(gr, 4, 3, 4);  
    imprime(gr);  
    int* r = dijkstra(gr, 0);  
}
```

Assinale a resposta com o menor caminho entre V0 e V2:

- a) V0 V2
- b) V0 V1 V2
- c) V0 V1 V4 V2**
- d) V0 V1 V3 V2
- e) V0 V1 V3 V4 V2

➤ **QUESTÃO 12:**

Considerando uma lista ligada cujo nó é:

```
typedef struct no {  
    int info;  
    struct no * proximo;  
} No;  
:
```

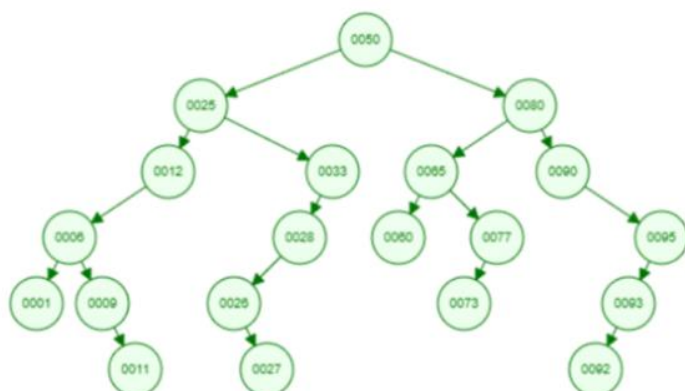
I	No* xx() { return NULL; }	A	Alocação da memória RAM com um nó
II	No* n_; n_ = (No*)malloc(sizeof(No));	B	Função de inicialização de uma lista ligada
III	No* lista; lista = xx();	C	Criação de uma lista ligada

Faça a associação:

- A) I-A ; II-B ; III-C
- B) I-B ; II-C ; III-B
- C) I-A ; II-C ; III-B
- D) I-C ; II-A ; III-B
- E) I-B ; II-A ; III-C

➤ **QUESTÃO 13:**

Considere a árvore binária de busca a seguir:



Ao remover os nós 25 e 80, assinale a alternativa com os valores que podem substituir-os mantendo a integridade da árvore binária de busca:

- a) 12, 33 e 65, 90
- b) 11, 26 e 77, 92
- c) 12, 27 e 73, 90
- d) 12, 26 e 77, 90
- e) 11, 27 e 73, 92

➤ **QUESTÃO 14:**

Um tipo nó foi definido com a seguinte estrutura:

```
typedef struct no { int info;  
struct no* próximo;  
}No;
```



Representado por:

Endereço de nó na memória RAM

Parte do TAD que manipula o nó segue a seguir:

```
typedef struct no {  
int info; struct no* próximo;  
}No;  
No* inicia() {  
Return NULL;  
}  
No* insere(No* lista, int num) {
```

```

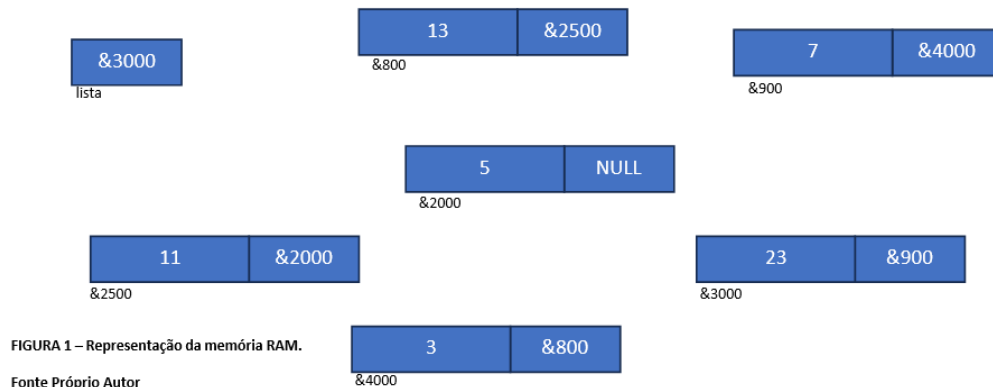
No* novo_no = (No*)malloc(sizeof (No)); novo_no ->info = num; novo_no->próximo = lista;
return novo_no;
}
void imprime(No* lista){ No* aux;
for (aux = lista; aux != Null; aux = aux->próximo)
printf(" %d", aux ->info);
}

```

Uma vez criada a lista com:

No* lista;

Em dado momento, temos a memória RAM representada conforme figura:



Assinale a alternativa com a saída ao executar a função `imprime(lista)`, com o valor do resultado:

- A) 5 11 13 3 7 23
- B) 13 7 5 11 23 3
- C) 3 23 11 5 7 13
- D) 23 7 3 13 11 5**
- E) 3 5 7 11 13 23

➤ QUESTÃO 15:

Partindo da memória RAM totalmente vazia, ao executar um programa na linguagem C assinale a alternativa que apresenta a sequência correta do carregamento da memória.

- A) Programa, variáveis estáticas, variáveis dinâmicas, área livre, e na ordem inversa o controle das funções**
- B) Variáveis estáticas e variáveis dinâmicas
- C) Variáveis dinâmicas e variáveis estáticas.
- D) Programa, variáveis dinâmicas, variáveis
- E) Programa, variáveis dinâmicas, variável e na ordem inversa, o controle das funções direta o controle das funções, área livre

➤ QUESTÃO 16:

Ao executar o programa a seguir:

```

1  #include <stdio.h>
2
3  void main(void){
4      int i=123;
5      int j=999;
6      printf("\nprintf1 %d", i);
7      printf("\nprintf2 %x", &i);
8      printf("\nprintf3 %d", j);
9      printf("\nprintf4 %x", &j);
10     int *ei=&i;
11     *ei=j;
12     printf("\nprintf5 %d", i);
13
14 }

```

Apresentou a seguinte saída a seguir:

```

printf1 123
printf2 61ff18
printf3 999
printf4 61ff14
printf5 999
PS K:\Estudos\Estrutura de Dados\output>

```

Os marcadores de comentário da linha 11 foram removidos, assinale a alternativa com a saída da linha 11.

- A) printf?999
- B) printf?98B72348
- C) printf?123
- D) printf?98b7234c
- E) Ao remover o comentário, resulta em erro.



QUESTÃO 17:

Ao executar o programa a seguir:

```

1  #include <stdio.h>
2- void teste(int a, int* b){
3      a++;
4      *b+=1;
5      a=a+ *b;
6  }
7- void main(void){
8      int a=1;
9      int b=5;
10     teste(a,&b);
11     printf("\na=%d b=%d",a,b);
12 }

```

Qual será a sua saída?

- A) a=1 b=5
- B) a=8 b=6
- C) a=8 b=5
- D) a=1 b=6**
- E) a=7 b=5



QUESTÃO 18:

Considere as seguintes estruturas:

```
typedef struct pessoa{  
    char nome[20];  
    int idade;  
} Pessoa;  
  
typedef struct sala{  
    Pessoa* primeiro;  
} Sala;
```

Considerando o conteúdo na memória RAM, que tipo de valor armazenará a variável “primeiro”?

- A) Estrutura de Sala
- B) Endereço de memória
- C) Estrutura Pessoa
- D) Cadeia de caracteres e um inteiro
- E) Vetor de caracteres e um inteiro.**



QUESTÃO 19:

Ao executar o programa abaixo:

```

#include <stdio.h>
#include <stdlib.h>
#define tamanho 12
char vetor[tamanho];
int nAtual = 0;

int remover(char vetor[], char valor, int n) {
    for (int posicao = 0; posicao < tamanho; posicao++) {
        if (vetor[posicao] == valor) {
            for (int j = posicao; j < tamanho; j++) {
                vetor[j] = vetor[j + 1];
            }
            vetor[tamanho - 1] = '\0';
            return --n;
        }
    }
    printf("Valor %c não localizado", valor);
    return n;
}

int inserir(char vetor[], char valor, int n) {
    if (tamanho <= n) {
        printf("A lista esta cheia");
        return n;
    }
    else {
        vetor[n] = valor;
        return ++n;
    }
}

void imprime(char vetor[], int n) {
    int posicao;
    printf("\n\nlista : \n\n");
    for (posicao = 0; posicao < n; posicao++)
        printf("%c | ", vetor[posicao]);
}

int main() {
    char valor;
    nAtual = inserir(vetor, 'h', nAtual);
    nAtual = inserir(vetor, 'i', nAtual);
    nAtual = inserir(vetor, 'p', nAtual);
    nAtual = inserir(vetor, 'o', nAtual);
    nAtual = inserir(vetor, 'p', nAtual);
    nAtual = inserir(vetor, 'o', nAtual);
    nAtual = inserir(vetor, 't', nAtual);
    nAtual = inserir(vetor, 'a', nAtual);
    nAtual = inserir(vetor, 'm', nAtual);
    nAtual = inserir(vetor, 'o', nAtual);
    valor = 'o';
    nAtual = remover(vetor, valor, nAtual);
    imprime(vetor, nAtual);
    return 0;
}

```

O programa é uma lista vetorial. Assinale a alternativa com a saída correta.

a.
| h | i | p | o | p | o | t | a | m | o | a |

b.
| h | i | p | p | t | a | m | a |

c.
| h | i | p | p | o | t | a | m | o | a |

d.
| h | i | p | a | p | o | t | a | m | o |

e.
| h | i | p | a | p | t | m |



QUESTÃO 20:

Temos o seguinte programa:

```
TAD - Filas
TAD Pilha
int main()
{
    int v[]={1,2,3,4};
    Pilha *p=criap();
    Fila * f=criaf();
    for (int i=0;i<4;i++)
        queue(f,v[i]);
    push(p,dequeue(f));
    push(p,dequeue(f));
    push(p,dequeue(f));
    push(p,dequeue(f));
    queue(f,pop(p));
    queue(f,pop(p));
    queue(f,pop(p));
    queue(f,pop(p));
    for (int i=0;i<4;i++)
        v[i]=dequeue(f);
}
```

Assinale a alternativa com a situação final do vetor v[]:

- A) {1,2,3,4}
- B) {1,2,2,1}
- C) {2,1,1,2}
- D) {4,3,3,4}
- E) {4,3,2,1}



QUESTÃO 21:

Considere o programa a seguir:

```
1 #include <stdio.h>
2 void MostraArea(float a){
3     printf("-----\n");
4     printf("Area\n");
5     printf("    %2.2f\n",a);
6     printf("-----\n");
7 }
8
9 void circulo(int raio){
10    float area,pi=3.14159;
11    area=pi*raio*raio;
12    MostraArea(area);
13    return area;
14 }
15 void quadrado(int lado){
16    int area=lado*lado;
17    MostraArea(area);
18    return area;
19 }
20 int main(){
21    int a;
22    scanf("%d",&a);
23    circulo(a);
24    quadrado(a);
25 }
```

Assinale a alternativa com as instruções que preenchem 1, 2 e 3 respectivamente para que o programa fique formalmente correto.

- A) Float, int, int
- B) Void, void, void
- C) Void, float, int
- D) Void, int, int
- E) Float, float, int



QUESTÃO 22:

Leia as seguintes frases:

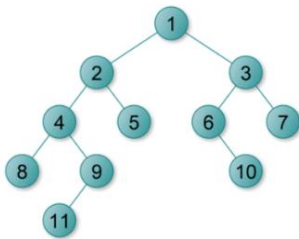
- Sequencia de passos que visam atingir a um objetivo definido.
- Descrição de uma sequência de passos que deve ser seguida para a realização de uma tarefa.
- Descrição de um conjunto de comandos que, obedecidos, resultam numa sucessão finita de ações. Definem qual conceito?

- A) Algoritmo
- B) Programa computacional
- C) Executável
- D) Estrutura de Dados
- E) Modularização



QUESTÃO 23:

Considere a árvore:



Assinale a alternativa com o percurso em ordem (infixo) e pós-ordem (pós-fixo):

- ☒ a. 8 4 11 9 2 5 1 6 10 3 7
8 11 9 4 5 2 10 6 7 3 1
- ☐ b. 8 4 11 9 5 2 1 10 6 3 7
8 4 9 11 5 2 1 3 6 10 7
- ☐ c. 8 4 11 9 2 5 1 6 10 3 7
8 4 9 11 5 2 1 3 6 10 7
- ☐ d. 8 4 11 9 5 2 1 10 6 3 7
8 11 9 4 5 2 10 6 7 3 1
- ☐ e. 1 2 4 8 9 11 5 3 6 10 7
8 11 9 4 5 2 10 6 7 3 1



QUESTÃO 24:

Temos o seguinte programa:

TAD – Fila:

```
int main()
{
    Fila * f=cria();
    queue(f, 10);
    queue(f, 9);
    queue(f, 8);
    dequeue(f);
    dequeue(f);
    queue(f, 7);
    queue(f, 6);
    dequeue(f);
    dequeue(f);
    queue(f, 5);
    queue(f, 4);
    dequeue(f);
    dequeue(f);
    queue(f, 3);
    printf ("\n%d\n",dequeue(f));
}
```

Assinale a alternativa com o valor apresentado na tela:

- a) 10
- b) 8
- c) 6
- d) 4
- e) 3



QUESTÃO 25:

1. Considerando o programa abaixo:

```
1 #include <stdio.h>
2 int main(){
3     int vetor[]=VETOR ;
4     int mem=vetor[0];
5     for (int i =9;i>0;i--){
6         if (mem==vetor[i]){
7             break;
8         }
9     }
10 }
```

Substituindo VETOR na declaração do programa, qual dos vetores abaixo apresenta o melhor caso na análise do programa acima?

- ☐ a. {1,1,2,3,4,5,6,7,8}
- ☒ b. {1,8,7,6,5,4,3,2,1}
- ☐ c. {9,8,7,6,5,4,3,2,1}
- ☐ d. {9,1,2,3,4,5,6,7,8}
- ☐ e. {0,1,2,3,4,5,6,7,8}



QUESTÃO 26:

- Sobre Deque, leia as afirmações abaixo:
- I- No Deque, a implementação necessária a ser feita utiliza uma lista duplamente encadeada.
 - II- No Deque, o nó descritor necessita de quatro ponteiros, o início e o fim da frente para trás, e o início e o fim de trás para frente.
 - III- No Deque, as operações de inserção (queue) e remoção (dequeue) da fila necessitam ser duplicadas, pois a fila agora movimenta para os dois sentidos.

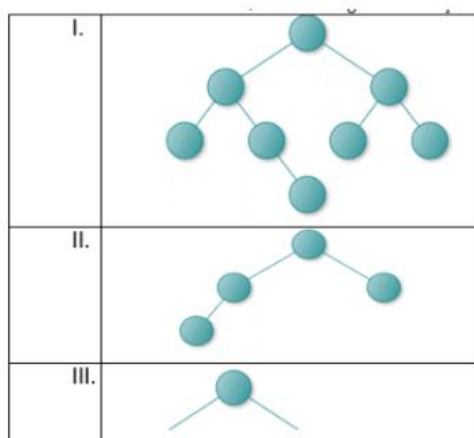
Assinale a alternativa correta

- ☐ a. Apenas a afirmação I está correta.
- ☐ b. Apenas a afirmação II está correta.
- ☒ c. Apenas a afirmação III está correta.
- ☐ d. Mais de uma das afirmações está correta.
- ☐ e. Nenhuma das afirmações está correta.



QUESTÃO 27:

1. Sobre altura de árvores binárias, temos as seguintes situações:



Assinale a alternativa com as alturas corretas de I, II e III respectivamente.

- ☐ a. 4,3,1.
- ☐ b. 4,3,0.
- ☐ c. 3,2,1.
- ☐ d. 4,2,1.
- ☒ e. 3,2,0.



QUESTÃO 28:

1. A árvore abaixo é uma árvore binária de busca:



Assinale a alternativa que não monta esta árvore:

- ☐ a. 8 4 12 2 7 6 10 11.
- ☐ b. 8 12 4 7 10 2 6 11.
- ☒ c. 8 4 2 7 6 12 11 10.
- ☐ d. 8 4 12 7 2 10 11 6.
- ☐ e. 8 12 10 11 4 7 2 6.

➤ QUESTÃO 29:

Sobre a Tabela Hash, afirma-se:

- I. Toda função hash utiliza o quociente da divisão para preencher a tabela.
- II. As colisões acontecem quando a função hash acha uma posição vaga.
- III. A função de hash, determina a posição na qual o elemento se encontra armazenado na tabela.

Assinale a alternativa correta em relação às afirmações acima:

- ☐ a. Apenas a afirmação I é correta.
- ☐ b. Apenas a afirmação II é correta.
- ☒ c. Apenas a afirmação III é correta.
- ☐ d. Mais de uma das afirmações está correta.
- ☐ e. Nenhuma das afirmações está correta.

➤ QUESTÃO 30:

I -	<p>A árvore das operações: expressão $a * (b + c) - d * e$</p>
II -	
III -	<p>Só existe um caminho entre dois nós</p>

Assinale a alternativa correta:

- A) Apenas a situação I é verdadeira.
- B) Apenas a situação II é verdadeira.
- C) Apenas a situação III é verdadeira.
- D) Mais de uma situação é válida.
- E) Nenhuma das situações é válida.

Resposta: (D) considerando que a expressão é $a \times (b - c) / d$



QUESTÃO 31:

1. Ao executar o programa abaixo:

```
1 #include <stdio.h>
2 #include<stdlib.h>
3
4 void main()
5 {
6     int tamanho = sizeof(int);
7     int *v, a=1;
8     v = (int *)malloc(10 * tamanho);
9     for (int i = 0; i < 10; i++)
10         v[i] = a++;
11     int *x = v + 2;
12     printf("\nValor armazenado em x é: %d\n", *x);
13 }
```

Assinale a alternativa com o valor o resultado:

- ☐ a. Valor armazenado em x é: 0
- ☐ b. Valor armazenado em x é: 1
- ☐ c. Valor armazenado em x é: 2
- ☒ d. Valor armazenado em x é: 3
- ☐ e. Valor armazenado em x é: 4



QUESTÃO 32:

Um programador desenvolveu o código abaixo para recriar os números de Fibonacci:

```
#include
```

```
int fibonacci(int num){
```

```
    if(num==1 || num==2)
```

```
        return 1;
```

```
    else
```

```
        return fibonacci(num-1) + fibonacci(num-2);
```

```
}
```

```
main(){
```

```
    int i;
```

```
    clrscr();
```

```
    for(i = 0; i < 5; i++)
```

```
printf("%d ", fibonacci(i+1));  
  
getch();  
  
}
```

Quando o ponto de retorno é alcançado?

- (A) Não existe condição de parada para este programa.
- (B) Quando laço for no método main chegar a 5.
- (C) Quando num alcança os valores um ou dois
- (D) Quando num somar 5
- (E) Quando num alcança os valores um e dois



QUESTÃO 33:

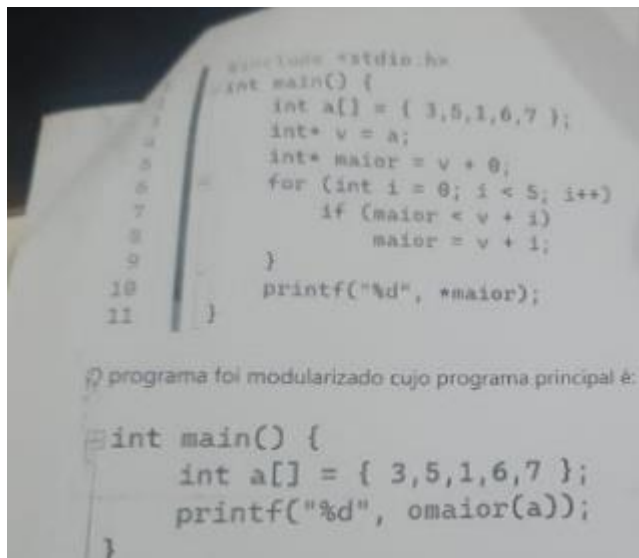
1. Sobre Tipo Abstrato de Dados, afirma-se:
- I. O conceito de tipo de dado abstrato depende do *hardware*.
 - II. TAD define o que cada operação faz, e como faz.
 - III. É uma especificação matemática de um conjunto de dados e das operações que podem ser executadas sobre esses dados.

Assinale a alternativa correta:

- ☐ a. Somente a afirmação I é verdadeira.
- ☐ b. Somente a afirmação II é verdadeira.
- ☒ c. Somente a afirmação III é verdadeira.
- ☐ d. Todas as afirmações estão corretas.
- ☐ e. Nenhuma das afirmações está correta.

QUESTÕES DISCURSIVAS:

1 - Considerando o programa a seguir:



Escreva a função omaior(...) mantendo a mesma lógica do programa sem modularização:

Resposta:

```

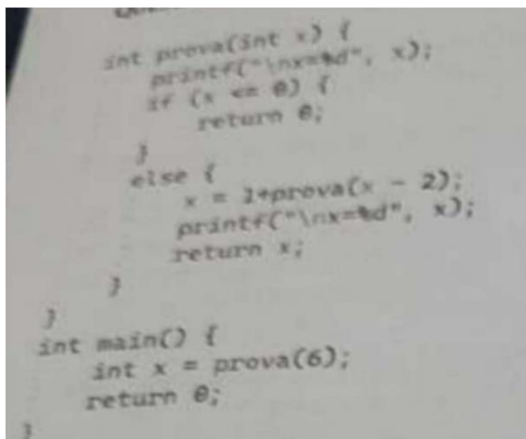
int omaior(int a[]) {
    int maior = a[0];

    for (int i = 1; i < 5; i++) {
        if (a[i] > maior) {
            maior = a[i];
        }
    }

    return maior;
}

```

2 - Considere o programa a seguir:



```

Int    prova(int x){
Printf(*\nx=&d" , x); If ( x<= 0) {
Return = 0;

```

```

}
Else {
X = 1+prova(x-2); Printf("\nx=&d", x); Return x
}
}
Int main (){
Int x = prova (6); Return 0;
}

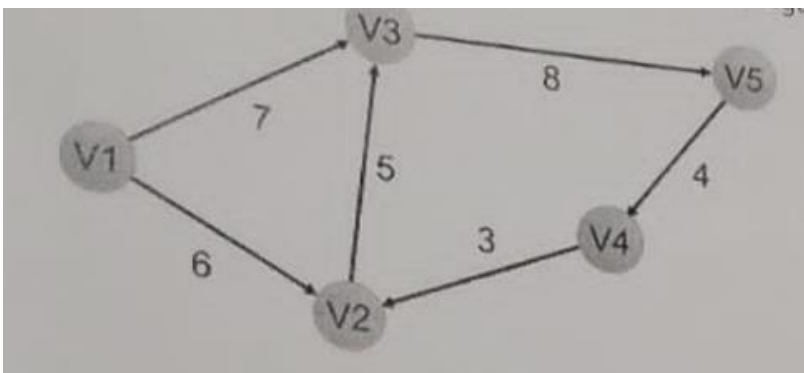
```

Escreva a saída deste programa:

Resposta:

x=6 x=4 x=2 x=0 x=1 x=2 x=3

3 – Monte a matriz de adjacências do grafo a seguir:



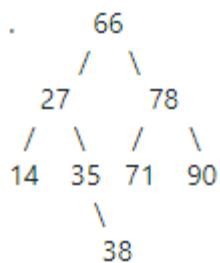
Resposta:

	V1	V2	V3	V4	V5
V1	0	6	7	0	0
V2	0	0	5	0	0
V3	0	0	0	0	8
V4	0	3	0	0	0
V5	0	0	0	4	0

4 – Monte uma árvore binária de busca com a seguinte sequência de entrada:

66, 27, 78, 90, 35, 14, 71, 38

Resposta:



5 – Considerando o programa a seguir, faça a modularização criando a função chamada módulo:

```

int main() {
    int a[4] = {3,5,1,2};
    int x, y, total;
    x = a[0];
    y = 0;
    for (int i = 1; i < x; i++) {
        y = y + i;
    }
    total = y;
    y = 0;
    for (int i = 1; i < x; i++) {
        y = y + i;
    }
    total = total + y;
    x = a[2];
}

```

```

y = 0;
for (int i = 1; i < x; i++) {
    y = y + i;
}
total = total * y;
x = a[3];
y = 0;
for (int i = 1; i < x; i++) {
    y = y + i;
}
total = total - y;
printf("Resultado=%d", total);
}

```

Resposta:

```

int modulo(int x){
    int y=0;
    for (int i=1;i<x;i++){
        y = y+i;
    }
    return y;
}

```

6 - Considere o programa incompleto a seguir:

Preencha corretamente o espaço para que o programa exiba corretamente as entradas em ordem crescente.


```

#include <stdio.h>
#include <string.h>
int main() {
    char c1[10], c2[10];
    scanf("%s", &c1);
    scanf("%s", &c2);
    if (strcmp(c1, c2) > 0)
        printf("crescente %s %s", c1, c2);
    else
        printf("crescente %s %s", c2, c1);
    return 0;
}

```

Resposta: strcmp(c2,c1)

7 – Crie um novo tipo de dado chamado Aluno formado pelos campos número (inteiro) e nota (float):

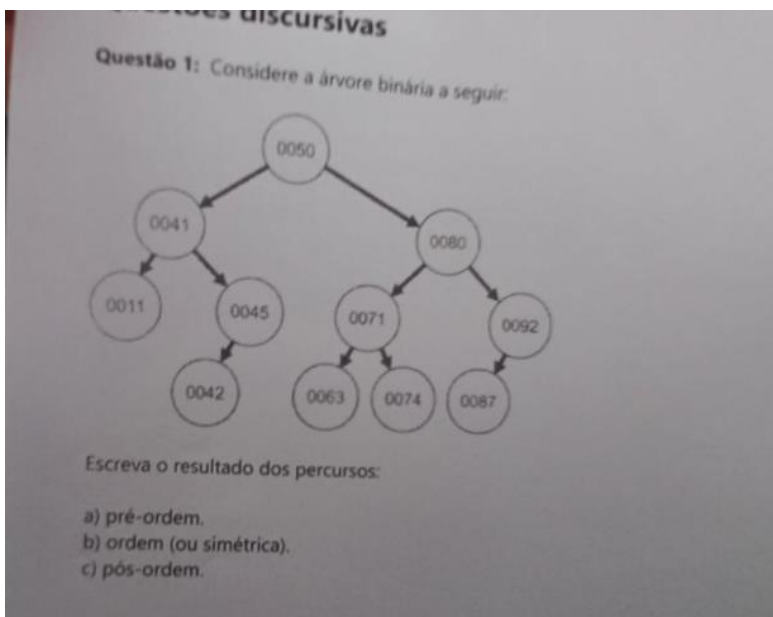
Resposta:

```

typedef struct Aluno {
    int numero;
    float nota;
} Aluno;

```

Questão 8 -



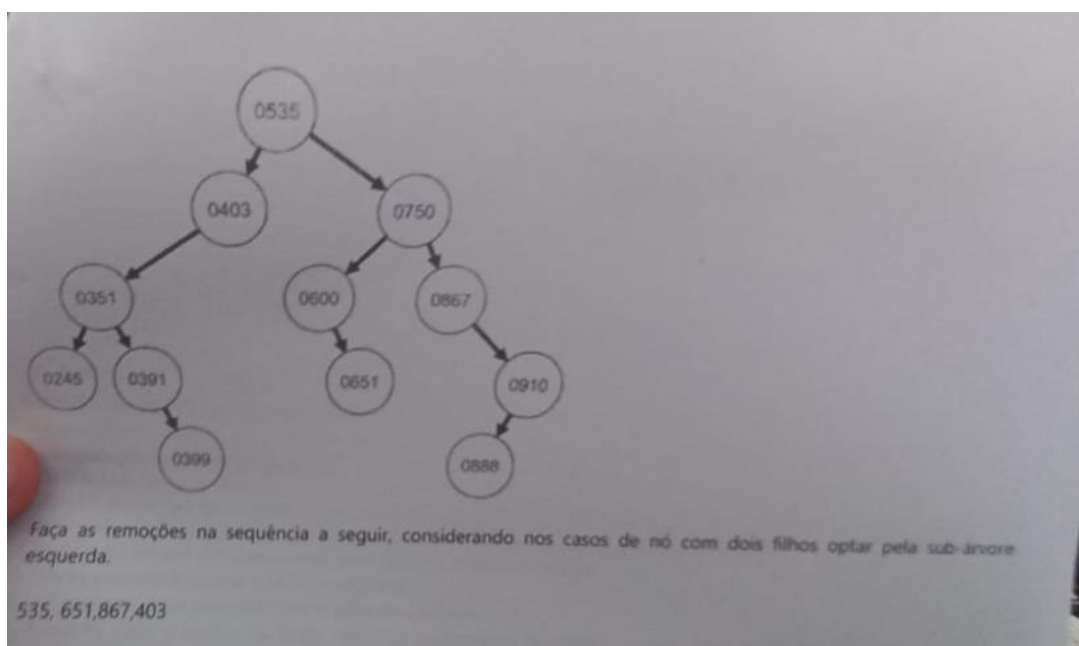
Pré-ordem: 50, 41, 11, 45, 42, 80, 71, 63, 74, 92, 87

Ordem: 11, 41, 42, 45, 50, 63, 71, 74, 80, 87, 92

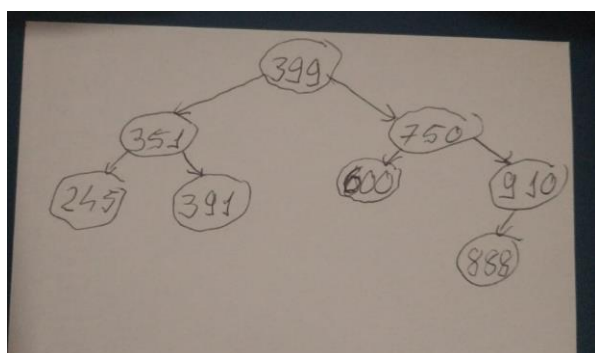
Pós- ordem: 11, 42, 45, 41, 63, 74, 71, 87, 92, 80, 50

a) pré-ordem:
 50, 41, 11, 45, 42, 80, 71, 63, 74, 92, 87
 b) ordem:
 11, 41, 42, 45, 50, 63, 71, 74, 80, 87, 92
 c) pós-ordem:
 11, 42, 45, 41, 63, 74, 71, 87, 92, 80, 50

Questão 9 -



Resposta:



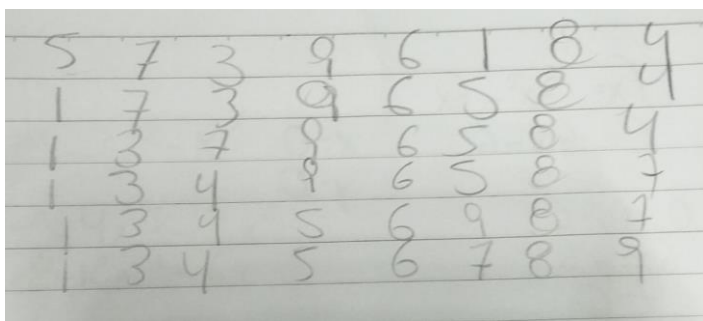
Questão 10 – A situação inicial do vetor v é:

V = 5 7 3 9 6 1 8 4

O vetor será ordenado pelo método de seleção (Selection Sort)

- Escreva a situação do vetor a cada rodada até estar totalmente ordenado.

Resposta:



A handwritten multiplication table on lined paper. The table consists of 6 rows and 8 columns of numbers. The numbers are written in a cursive, handwritten style. The first row contains the numbers 5, 7, 3, 9, 6, 1, 8, 4. The second row contains 1, 7, 3, 9, 6, 5, 8, 4. The third row contains 1, 3, 7, 9, 6, 5, 8, 4. The fourth row contains 1, 3, 4, 9, 6, 5, 8, 7. The fifth row contains 1, 3, 4, 5, 6, 9, 8, 7. The sixth row contains 1, 3, 4, 5, 6, 7, 8, 9.

5	7	3	9	6	1	8	4
1	7	3	9	6	5	8	4
1	3	7	9	6	5	8	4
1	3	4	9	6	5	8	7
1	3	4	5	6	9	8	7
1	3	4	5	6	7	8	9