

Aluno(a): .....

1. O que será impresso pelo programa abaixo?

```
#include <stdio.h>

int main() {
    int x, *y, z;
    x = 2;
    y = &x;
    (*y)++;
    z = (x++)*2;
    printf("%d,%d,%d",x,*y,z);
}
```

Resposta: .....

2. No programa a seguir, a variável *p* deve apontar para um bloco de memória que armazena os 10 primeiros múltiplos de 3 (isto é, o resultado de  $3 \times 1, \dots, 3 \times 10$ ). Preencha as lacunas para que o programa preencha esse bloco de memória adequadamente (linhas 4 e 5) e imprima o conteúdo armazenado nele.

```
1 int main(){
2     int i,*p;
3     p = _____;
4     for(_____) {
5         _____;
6     }
7     for(i=0;i<10;i++){
8         printf("%d ", ____);
9     }
10 }
```

Linha 3: .....

Linha 4: .....

Linha 5: .....

Linha 8: .....

3. Complete as lacunas da função a seguir para que ela faça a inserção de um elemento ao fim da lista.

Assuma que a lista *não* está vazia.

```
1 typedef struct lstItem{
2     int dado;
3     struct lstItem *next;
4 }listaItem;
5
6 listaItem *ins_fim(listaItem *lista,
7                     int dado){
8     listaItem *novo = _____;
9     novo->dado = dado;
10    novo->next = NULL;
11    if(lista==NULL){
12        lista = novo;
13    }
14    else{
15        listaItem *ultimo = lista;
16        while(_____){
17            ultimo = ultimo->next;
18        }
19        _____ ;
20    }
21    return lista;
```

4. Considere as seguintes estruturas, utilizadas para implementar uma fila:

```
typedef struct{
    int dado;
    struct item *next;
}item;

typedef struct{
    item *inicio;
    item *fim;
}fila;
```

Complete o código abaixo para implementar a função *push* em uma fila, assumindo que o parâmetro *\*f* é um ponteiro para o primeiro elemento da estrutura.

```
1 void push(fila *f, int dado){
2     item *novo =
3         malloc(sizeof(item));
4     novo->dado = dado;
5     novo->next = NULL;
6     if(f->fim!=NULL)
7         -----;
8     else
9         -----;
10    ----- = novo;
11 }
```

5. Considere uma tabela hash de tamanho *t* que armazena chaves cujos valores cujas chaves são números inteiros. O cálculo do hash para uma chave *k* é dado por  $h(k) = k \% t$  (método da divisão). Assumindo que a tabela está inicialmente vazia, qual será o número de colisões caso sejam inseridos elementos com as chaves 24, 28, 30, 36 e 40 em tabelas com os seguintes tamanhos:

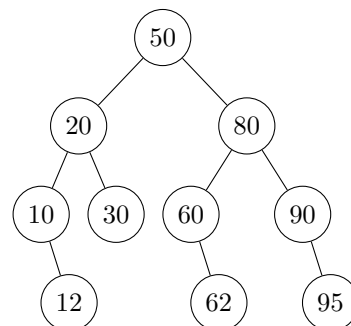
a)  $t = 3$ : .....

b)  $t = 4$ : .....

c)  $t = 7$ : .....

d)  $t = 12$ : .....

6. Considere a árvore a seguir:



Escreva abaixo a sequência de números que será impressa caso a árvore seja percorrida

a) Em pré-ordem: .....

b) Em ordem: .....

c) Em pós-ordem: .....

7. Desenhe a árvore AVL gerada pela inserção, em sequência, dos seguintes números: 25, 99, 91, 38, 18, 30, 50, 65, 20, 42, 57.

