Universidade Federal de Santa Catarina - Campus Blumenau
Departamento de Engenharia de Controle, Automação e Computação **Avaliação - REC**
BLU3202 — **Algoritmos e Estruturas de Dados** 2019/2

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 abaixo, o usuário informa o tamanho de um vetor a ser criado. Esse tamanho é armazenado na variável `q` . O vetor deve armazenar os "q" múltplos de 5 ($5 \times 1, 5 \times 2, \cdots, 5 \times q$). Em seguida o programa deve imprimir os elementos do vetor. Preencha as lacunas abaixo para que o programa tenha esse comportamento.

```
1  int main(){
2     int *v, q, i;
3     printf("Tamanho do vetor: ");
4     scanf("%d",&q);
5     v = ______________________;
6     for(int i=0;i<q;i++){
7        _______ = (q+1)*5;
8     }
9     for(i=0;i<q;i++){
10       printf("%d\n", _____);
11    }
12 }
```

Linha 5: _____
Linha 7: _____
Linha 10: _____

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;
```

Linha 8: _____
Linha 16: _____
Linha 19: _____

4. Considere a seguinte sequência de operações:
`push(9), push(3), pop(), push(2), pop(), push(4), push(7), pop(), pop(), push(6), push(1), pop(), push(2), push(5), pop()`.

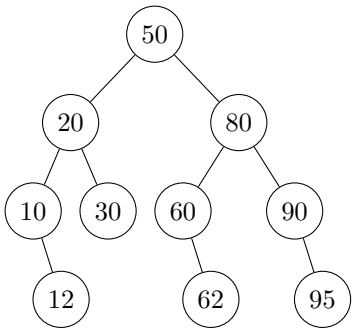
   - A soma dos elementos remanescentes em uma *fila*, inicialmente vazia, após essa sequência de operações será _____.
   - A soma dos elementos remanescentes em uma *pilha*, inicialmente vazia, após essa sequência de operações será _____.

5. Considere uma tabela hash com 23 espaços indexados de 0 a 22, utilizando a função hash $h(x) = x \ mod \ 23$ (método da divisão). Nesta tabela, são inseridas, na ordem dada, as seguntes chaves: 44, 46, 49, 70, 27, 71, 90, 97, 95. Responda:

   (a) Quais chaves serão envolvidas em colisão?
   _____

   (b) Esboce a tabela hash após a inserção das chaves. Utilize o formato `[indice]:{chaves}`.
   _____
   _____
   _____
   _____
   _____
   _____
   _____
   _____

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: 20, 30, 10, 25, 35, 28, 23, 22, 38, 21.

8. Considere o vetor $v = [2, 6, 18, 19, 24, 27, 37, 43, 56, 99]$. Escreva a sequência de elementos avaliados na busca binária pelos seguintes números:
   - 19: _____
   - 37: _____
   - 47: _____

9. Escreva o estado do vetor $v = [74, 91, 5, 3, 9, 43, 55, 71, 58, 20]$ após cada um dos passos da execução do algoritmo de ordenação por seleção.

10. Aplique o algoritmo de ordenação *Quick sort* no vetor $v = [30, 68, 36, 49, 61, 58, 94, 78]$ utilizando como pivô p o elemento central do vetor (ou seja, sendo l e r os índices das extremidades esquerda e direita do vetor respectivamente, considera-se $p = \lfloor \frac{l+r}{2} \rfloor$). Durante a ordenação, o subvetor mais à esquerda do pivô deve ser ordenado antes do subvetor mais à direita. Mostre cada um dos passos da ordenação que levaram a obter o vetor ordenado. Considere que um *passo de ordenação* está completo quando o pivô está em sua posição definitiva.

Informações úteis

- Em vetores com número par de elementos, considerar, como elemento central, o último elemento da primeira metade.
- Quando dois subvetores precisarem ser ordenados, considerar que o subvetor da esquerda é ordenado antes do subvetor da direita.
- Ao dividir um vetor $v = [v_0, \cdots, v_n]$ pela metade, sendo 0 (zero) o índice do primeiro elemento e n o índice do último elemento, considerar que (i) a primeira metade é $[v_0, \cdots, v_c]$ e (ii) a segunda metade é $[v_{c+1}, \cdots, v_n]$, onde $c = \lfloor \frac{0+n}{2} \rfloor$.
- Em um algoritmo de ordenação, um *passo completo* acontece quando um determinado número do vetor é colocado em sua posição apropriada. Por exmplo, considerando o número 90 no vetor $v = [90, 50, 30]$, ao final de um passo completo, o vetor teria a seguinte ordem: $v = [50, 30, 90]$.