

ATIVIDADE DE FIXAÇÃO 1

Informações importantes:

- 1. Os exercícios que não pedirem a implementação de código favor descrever a resposta e anexar como comentário no código fonte.
- 2. Em todos os exercícios, caso entenda necessário ou que fique melhor, vocês podem fazer mais de um predicado para resolver o problema.
- 3. Para todos os exercícios vocês devem fazer um comentário no código explicando como executar o código citando exemplos.
- 4. Para todas as funções recursivas devem ser descritas o seu funcionamento.

```
1)
void altera1(int **p, int *a)
{
    **p = *a;
    *a = *a +50;
}

void altera2(int **p, int *b)
{
    *p = b;
    *b = *b +30;
}

int main()
{
    int x,y, *px, *py;

    x = 10;
    y = x + 20;

    px = &x;
    py = &y;

    printf("x = %d, End. x = %p, px = %p, y = %d \n",x,&x, px,y);
    altera1(&px, &y);
    printf("x = %d, End. x = %p, px = %p, y = %d \n",x,&x, px,y);
    getchar();

    printf("y = %d, End. y = %p, py = %p, x = %d \n",y,&y, py,x);
    altera2(&py, &x);
    printf("y = %d, End. y = %p, py = %p, x = %d \n",y,&y, py,x);
    getchar();

    return(0);
}
```

Dado o código acima, responda as seguintes questões:

- (a) Qual a diferença entre px e x?
- (b) Qual a diferença entre px e py?
- (c) Quais são os valores impressos pelo primeiro printf?
- (d) O que muda do primeiro printf para o segundo?
- (e) Quais os valores impressos pelo terceiro print?
- (f) O que muda do terceiro para o quarto print?
- (g) Explique a diferença entre o altera1 e o altera2.

- 2) Faça uma função que dado um número inteiro devolva os divisores do mesmo, a função deve ser recursiva.
- 3) Faça uma função em C que leia um vetor de strings não ordenado e depois faça:
  - (a) uma função recursiva com pendência que devolva a string de maior tamanho.
  - (b) uma função recursiva sem pendência que devolva a quantidade de strings que iniciam com vogal.
  - (c) uma função recursiva que devolva um vetor contendo somente as strings com tamanho  $\geq 4$  e que iniciam com letra maiúscula.
- 4) Faça um algoritmo em C que leia 4 números inteiros e então faça:
  - (a) Uma função recursiva sem pendência que calcule Máximo Divisor Comum (MDC) entre os números lidos.
  - (b) Repita o exercício da letra a, mas agora faça uma função recursiva que deixa pendência.
- 5) Dado uma lista simplesmente encadeada e ordenada de pessoas:

```
struct pessoa{
    int Id;
    String Nome;
    int AnoNascimento;
    int Altura;
};

struct lista{
    Struct pessoa Info;
    Struct lista *Prox;
};
```

Faça as seguintes funções:

- (a) Uma função recursiva que devolva o nome da pessoa mais velha da lista, caso exista mais de uma devolva um vetor com os nomes.
- (b) Uma função recursiva que devolva a menor e a maior altura.
- (c) Uma função recursiva que devolva o número de pessoas com a altura mediana, utilize o resultado do item (b) para calcular a altura mediana.
- (d) Uma função recursiva que devolva o número de pessoas abaixo de 40 anos com a altura maior do que a altura mediana.