



Universidade Federal do Maranhão
Departamento de Informática
Ciência da Computação
Linguagem de programação I
Prof.: Francisco Glaubos

Prova 1 - Tipo 2

2023.2

- 1) (5,0 pts) Implemente uma função recursiva em C que calcule e retorne a soma de todos os elementos de um array de inteiros. A função deve receber o array ($0 < |\text{array}| \leq 100$) e o seu tamanho. A função deve ser implementada sem utilizar estruturas de repetição (loops) e sem utilizar variáveis globais ou estáticas. A assinatura da função deve ser:

```
int findsum(int a[], int n);
```

Exemplos:

```
int arr[] = {-1, -2, -3, -4, -5};  
Soma: -15
```

```
int arr[] = {-1, 2, -3, 4, 0};  
Soma: 2
```

- 2) (5,0 pts) Crie uma função em C que receba três argumentos: um array `precos[i]` que representa o preço de uma determinada ação no i -ésimo dia, um array de `bonus[i]` que representa um bônus de

venda que você recebe no i -ésimo dia além do preço da ação, e um inteiro $n : n \leq 100$ que representa o número de dias.

Você deseja maximizar seu lucro escolhendo um único dia para comprar uma ação e escolher um dia diferente no futuro para vender essa ação. Retorne o lucro máximo que você pode obter com esta transação. Se você não conseguir nenhum lucro, retorne 0. Você deve implementar a seguinte função, e imprimir na tela o dia da compra e o dia da venda

Obs.: Utilize pelo menos uma diretiva de compilação e uma classe de armazenado

```
int maxProfit(int precos[], int  
bonus[], int n)
```

Exemplo:

Entrada:

```
int precos[] = {7, 1, 5, 3, 6, 4};  
int bonus[] = {0, 0, 1, 0, 2, 0};
```

Saída (em tela):

Dia de Compra: 1

Dia de Venda: 4

O Lucro Máximo é: 7

1) #include <stdio.h>

~~int findUmn~~

int findUmn(int a[], int n){

 int rama, alre;

 if (n == 1){

 rama = a[n-1];
 alre =

 alre = findUmn(a, n-1);

 rama = a[n-1] + alre;

 return rama;

}

31

está fora do bloco

do else?

int main(){

 int arr[5] = {-1, -2, -3, -4, -5};

 int n = 5;

 int rumn = findUmn(arr, n);

 printf("o valor da subseqüência é: %d \n", rumn);

 return 0;

}

