

Exercícios



Arquivos Cabeçalhos e Bibliotecas Externas

Para ajudar na organização do código criamos uma **biblioteca de funções** e um **arquivo cabeçalho** de declaração, deixando apenas o **main** no arquivo principal.

Arquivos Cabeçalhos e Bibliotecas Externas

Biblioteca de funções: arquivo com extensão `.c` contendo funções referentes a um certo “tema”. Ex.:

`math.c` - biblioteca com funções matemáticas

`string.c` - biblioteca para manipulação de strings

Arquivos Cabeçalhos e Bibliotecas Externas

Arquivo cabeçalho: arquivo com extensão `.h` contendo a declaração das funções de uma biblioteca. Ex.:

`math.h` - declaração das funções matemáticas, tipos de entrada e saída esperados.

Você deve incluir esse arquivo no código principal.

Compilando

Arquivos:

matematica.c

matematica.h

cabecalho.c

```
gcc -o cabecalho cabecalho.c matematica.c
```

Segmentation Fault

Causa: leitura ou escrita em determinada área de memória sem a devida permissão ou alocação.

Exemplo trivial:

```
char * x = NULL;
```

```
*x = 10; ← x não tem espaço de memória alocado
```

Evitando Segmentation Fault

Modularização: ajuda na detecção do bug

The printf approach: imprima passos e saídas intermediárias nas regiões suspeitas

Suspeite dos operadores & e *: em muitos casos as linhas contendo & e * são as culpadas

Ex 01 - Concatenação

Faça um algoritmo que receba duas strings: `char * s1`, `char * s2` e concatene-as.

Para isso é necessário determinar o tamanho delas!

Ex 02 - Primos

Crie um algoritmo que preencha uma array chamada **primo** do tipo char com N elementos, sendo que o valor de $\text{primo}[i] = 1$ se i for primo e 0, caso contrário.

Imprima a soma dos elementos dessa array.

Ex 03 - Crivo de Eratosthenes

Dada uma array com todos os valores iguais a 1, começando da posição 2, podemos marcar todos os múltiplos de 2 com 0.

O próximo elemento igual a 1 será o 3, repetimos com os múltiplos de 3, e assim por diante.

Não é necessário fazer operações de divisão.

Imprima a soma dos elementos dessa array.

Compare com o tempo de execução do algoritmo anterior.