UNIVERSIDADE FEDERAL DO CEARÁ

DEPARTAMENTO DE COMPUTAÇÃO

CK0179 – Programação Computacional para Engenharia:
Criando Bibliotecas

Prof. Maurício Moreira Neto

Objetivo

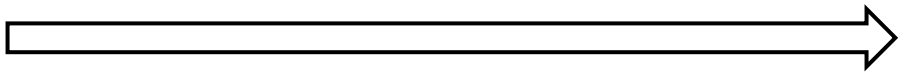
- Aprender a criar bibliotecas com funções próprias usando a linguagem C

- A partir disso, é possível criar bibliotecas próprias para utilização futura

Relembrando

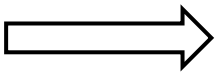
- Vamos relembrar como os códigos são feitos até então...

```
#include<stdio.h>
#include<stdlib.h>
```



Diretivas de include (bibliotecas)

```
void somar(float valor1, float valor2, float *r);
void subtrair(float valor1, float valor2, float *r);
void multiplicar(float valor1, float valor2, float *r);
void dividir(float valor1, float valor2, float *r);
```



Declarações de funções

Relembrando

```
int main(){
        float a, b, res = 0;
        int op;
        printf("Numero 1: ");
        scanf("%f", &a);
        printf("Numero 1: ");
        scanf("%f", &b);
        printf("Escolha opcao de operacao: ");
        scanf("%d", &op);
        switch (op){
                case 1: somar(a, b, &res); break;
                case 2: subtrair(a, b, &res); break;
                case 3: multiplicar(a, b, &res); break;
                case 4: dividir(a, b, &res); break;
                default: printf("Nenhuma operacao foi
escolhida");
        }
        printf("O resultado da operacao foi: %.2f", res);
        return 0;
}
```

Função principal que normalmente chama as funções declaradas anteriormente

Relembrando

```
void somar(float valor1, float valor2, float *r){
        *r = valor1 + valor2;
}
void subtrair(float valor1, float valor2, float *r){
        *r = valor1 - valor2;
}
void multiplicar(float valor1, float valor2, float *r){
        *r = valor1 * valor2;
}
void dividir(float valor1, float valor2, float *r){
        if (valor2 == 0){
                printf("Nao pode divisao por zero!");
        } else{
                *r = valor1/valor2;
        }
}
```

Implementação (definição) das funções declaradas anteriormente

Criando a Bibliotecas

- Regras para criação de arquivos de cabeçalhos:

1. Este arquivo deve conter apenas as declarações que serão visíveis ao programa
2. Deve conter apenas as diretivas #include necessárias para suas declarações
3. Proteja-o de declarações duplicadas usando uma guarda de cabeçalho
4. Nomeie corretamente o símbolo de seu arquivo de cabeçalho.

- Vamos criar um arquivo chamado **calculadora.h**

Criando a Bibliotecas

- Arquivo **calculadora.h**

```
#ifndef CALCULADORA_HEADER_
#define CALCULADORA_HEADER_

void somar(float valor1, float valor2, float *r);
void subtrair(float valor1, float valor2, float *r);
void multiplicar(float valor1, float valor2, float *r);
void dividir(float valor1, float valor2, float *r);

#endif
```

Criando a Bibliotecas

- A primeira linha verifica se o símbolo **CALCULADORA_HEADER_** está definido
- Caso não esteja definido, o conteúdo até **#endif** será incluído pelo compilador
- **CALCULADORA_HEADER_** é definido dentro desse intervalo, o que impede que o compilador duplique as duas declarações logo abaixo.

Criando a Bibliotecas

- **O que deve ser evitado:**
  - Com dois sublinhados consecutivos (**por__exemplo**)
  - Que comece com um sublinhado seguido por uma letra maiúscula (**_CALCULADORA_HEADER_** é reservado pela definição da linguagem)
  - Que comece com um sublinhado no contexto global (**_calculadora_header_** é reservado, pois está no contexto global)

Criando a Bibliotecas

- Note que a regra para nomenclatura é simples:
  - o nome do arquivo em maiúsculas + **_HEADER_**

Criando a Bibliotecas

- Agora que temos o arquivo com as declarações das funções, é necessário criar as definições destas funções (sua implementação)

- Crie um arquivo um **calculadora.c** com as definições com das funções declaradas em **calculadora.h**
  - **Importante:** o nome é o mesmo, só muda a extensão

Criando a Bibliotecas

- As regras para criar o arquivo .C são:

1. Deve conter apenas as definições, que não serão visíveis ao programa

2. Deve conter apenas as diretivas **#include** necessárias para suas definições

Criando a Bibliotecas

Arquivo **calculadora.c**

```
#include<stdio.h>
#include<stdlib.h>

void somar(float valor1, float valor2, float *r){
          *r = valor1 + valor2;
}
void subtrair(float valor1, float valor2, float *r){
          *r = valor1 - valor2;
}
void multiplicar(float valor1, float valor2, float *r){
          *r = valor1 * valor2;
}
void dividir(float valor1, float valor2, float *r){
          if (valor2 == 0){
                    printf("Nao pode divisao por zero!");
          } else{
                    *r = valor1/valor2;
          }
}
```

Criando a Bibliotecas

- Ainda não acabou...
- Provavelmente isso irar gerar um erro!

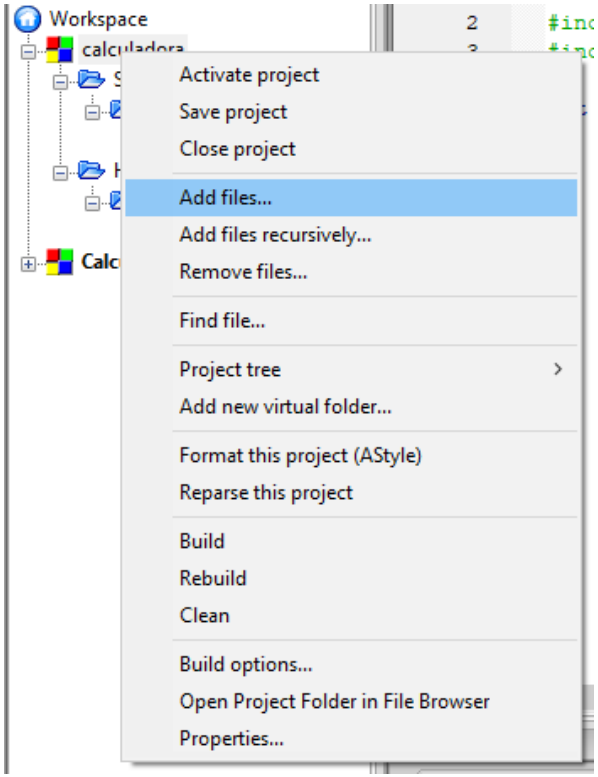
- Isso porque os arquivos **calculadora.h** e **calculadora.c** não foram unidas em um projeto para gerar um arquivo de extensão **.a**

- **Vamos ver o passo-a-passo para fazer isso no Code::Blocks**

Importante: isso depende da IDE!!

Criando a Bibliotecas

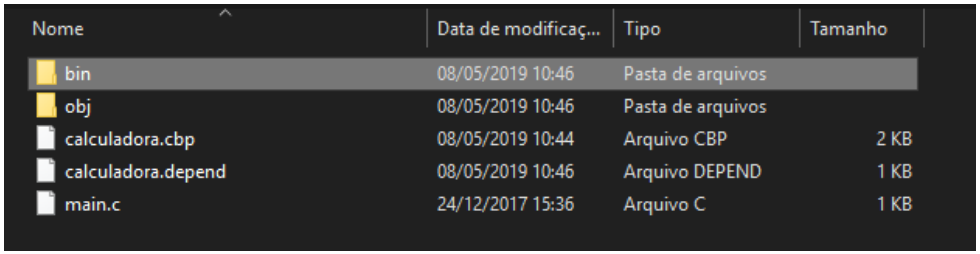
- Crie um novo projeto → **Static Library**
- Como título do projeto, coloque **calculadora**
- Agora adicione os arquivos **.h** e **.c** no projeto de biblioteca estática
- Exclua o arquivo **main.c** do projeto da biblioteca estática
- Por fim, gere o **build** deste projeto!



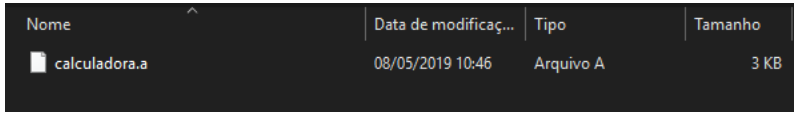
Criando a Bibliotecas

- Ao “buildar” o projeto de biblioteca estática, irá aparecer um arquivo novo chamado **libcalculadora.a**

- Deve-se renomear este arquivo para: **calculadora.a**



Nome	Data de modificaç...	Tipo	Tamanho
bin	08/05/2019 10:46	Pasta de arquivos	
obj	08/05/2019 10:46	Pasta de arquivos	
calculadora.cbp	08/05/2019 10:44	Arquivo CBP	2 KB
calculadora.depend	08/05/2019 10:46	Arquivo DEPEND	1 KB
main.c	24/12/2017 15:36	Arquivo C	1 KB



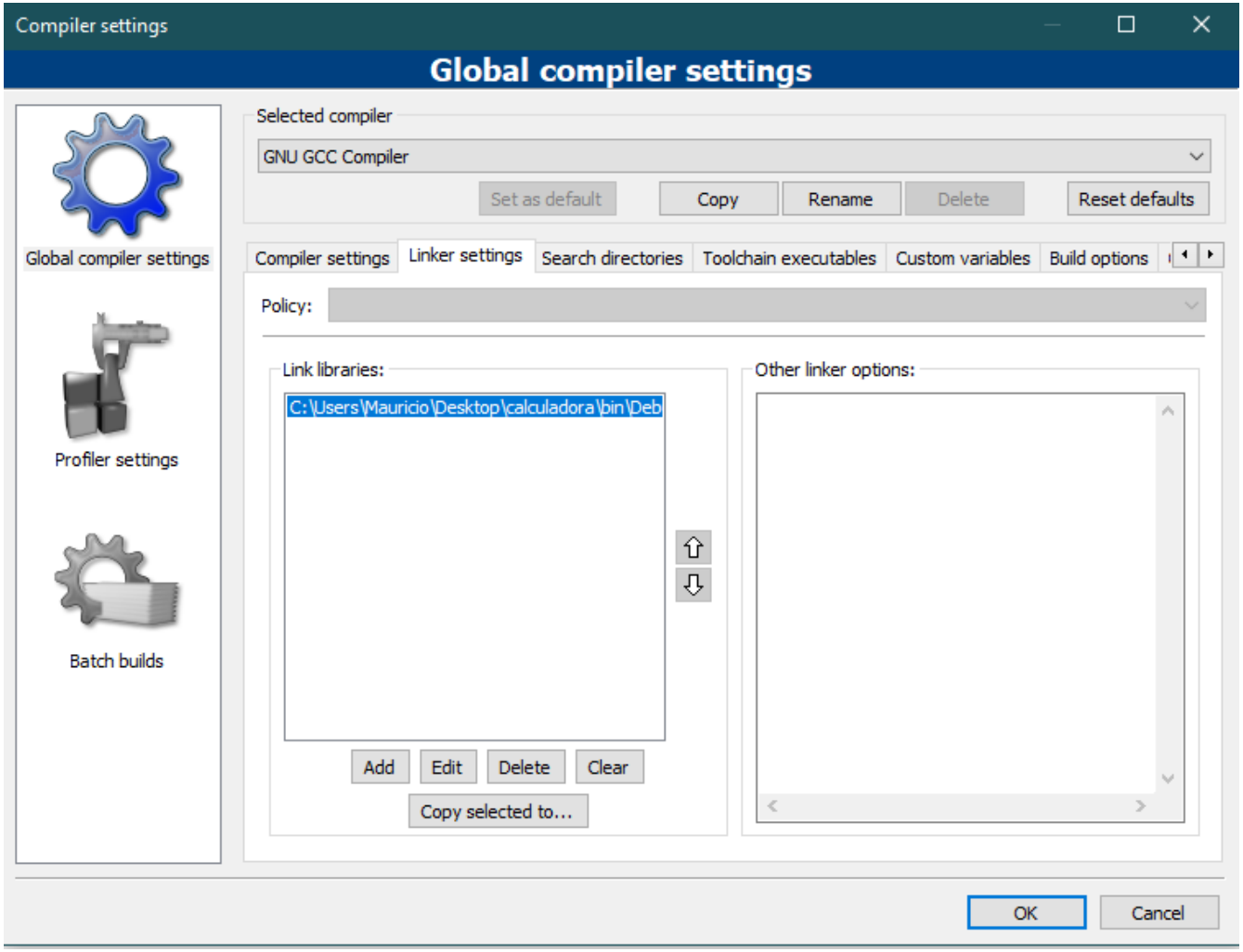
Nome	Data de modificaç...	Tipo	Tamanho
calculadora.a	08/05/2019 10:46	Arquivo A	3 KB

Criando a Bibliotecas

- Voltando para nosso projeto Console Application, faremos o seguinte passo-a-passo:

- Va em **Settings → Compiler → Linker Settings**

→ **Add** (após isso, deve-se colocar o arquivo **calculadora.a** que está no projeto da biblioteca estática)


Compiler settings
Global compiler settings
Global compiler settings
Profiler settings
Batch builds
Selected compiler
GNU GCC Compiler
Set as default
Copy
Rename
Delete
Reset defaults
Compiler settings
Linker settings
Search directories
Toolchain executables
Custom variables
Build options
Policy:
Link libraries:
C:\Users\Mauricio\Desktop\calculadora\bin\Deb
Other linker options:
Add
Edit
Delete
Clear
Copy selected to...
OK
Cancel

Dúvidas?