

```
#ifndef AULA_H  
#define AULA_H
```



Bibliotecas / Modularização

Quero poder reaproveitar as
funções que eu já fiz.

Quero poder compartilhar as
funções que fiz.

Biblioteca

Quero poder reaproveitar as
funções que eu já fiz.

Quero poder compartilhar as
funções que fiz.



**Quem são os
livros?**

Funções

**Uma biblioteca é: um
lugar onde funções ficam
guardadas**

Muito bonito!

Como é isso em C?

**Quero fazer funções que
resolvem limites,
derivadas e integrais**

programa.c

```
int main()  
...
```

Chama

```
resolver_limite(limite);
```

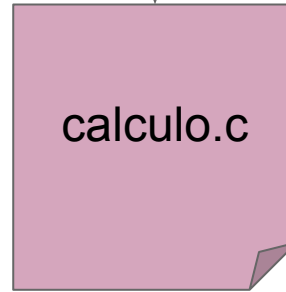
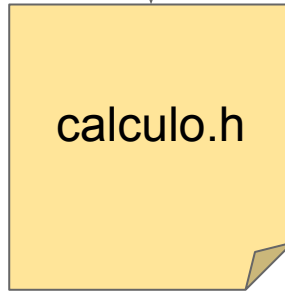
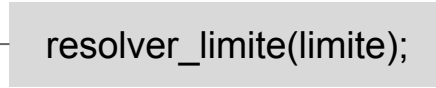
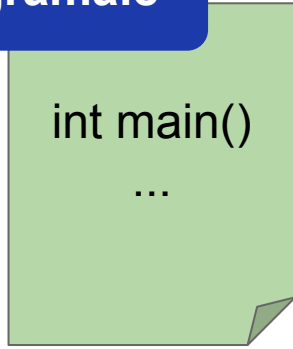
```
resolver_derivada(derivada);
```

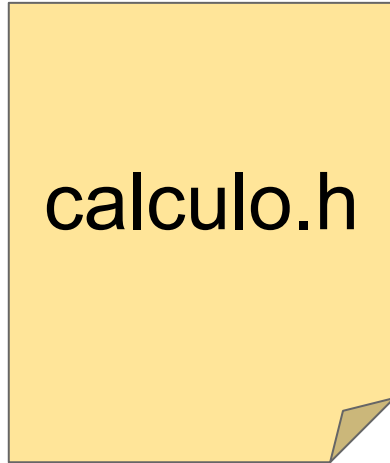
Pertencem

calculo.h

calculo.c

Biblioteca de Calculo





Arquivo de cabeçalho (Header file)

Guarda a

DEFINIÇÃO/ASSINATURA das
funções daquela biblioteca

Assinatura da Função

```
int soma(int a, int b) {  
    return a + b;  
}
```

Função
implementada

```
int soma(int a, int b);
```

Assinatura

Assinatura da Função

```
int soma(int a, int b) {  
    return a + b;  
}
```

Função
implementada

Termina com ponto e
vírgula

```
int soma(int a, int b);
```

Assinatura



calculo.c

Código fonte (Source file)

Guarda a **IMPLEMENTAÇÃO** das
funções daquela biblioteca

Implementação da Função

```
int soma(int a, int b) {  
    return a + b;  
}
```

Função
implementada

```
int soma(int a, int b);
```

Assinatura

Definições

calculo.h

Implementações

calculo.c

Biblioteca

```
graph TD; A[calculo.h] --- B[calculo.c]; B --> C[(Biblioteca)];
```

The diagram illustrates the relationship between header and source files and a library. It features two columns: 'Definições' (Definitions) on the left and 'Implementações' (Implementations) on the right. Under 'Definições' is a yellow document icon labeled 'calculo.h'. Under 'Implementações' is a purple document icon labeled 'calculo.c'. A horizontal line connects the two documents. A vertical arrow points from the center of this line down to a light blue cylinder labeled 'Biblioteca'.

Tá -.-

Como é isso em C?

calculo.h

```
#ifndef CALCULO_H
#define CALCULO_H

int soma(int a, int b);

#endif
```

calculo.c

```
#include "calculo.h"

int soma(int a, int b) {
    return a + b;
}
```

```
#include <stdio.h>
#include "calculo.h"

int main() {
    printf("%d\n", soma(1, 4));
    return 0;
}
```

programa.c

calculo.h

```
#ifndef CALCULO_H  
#define CALCULO_H
```

```
int soma(int a, int b);
```

```
#endif
```

calculo.h

```
#ifndef CALCULO_H  
#define CALCULO_H
```

```
int soma(int a, int b);
```

```
#endif
```

Include guards

Macros

calculo.h

```
#ifndef CALCULO_H  
#define CALCULO_H
```

```
int soma(int a, int b);
```

```
#endif
```

Assinaturas das
funções

calculo.c

```
#include "calculo.h"

int soma(int a, int b) {
    return a + b;
}
```

Inclusão do
header file

Obrigatório

calcula.c

```
#include "calcula.h"
```

```
int soma(int a, int b) {  
    return a + b;  
}
```

Implementação
das funções

calcula.c

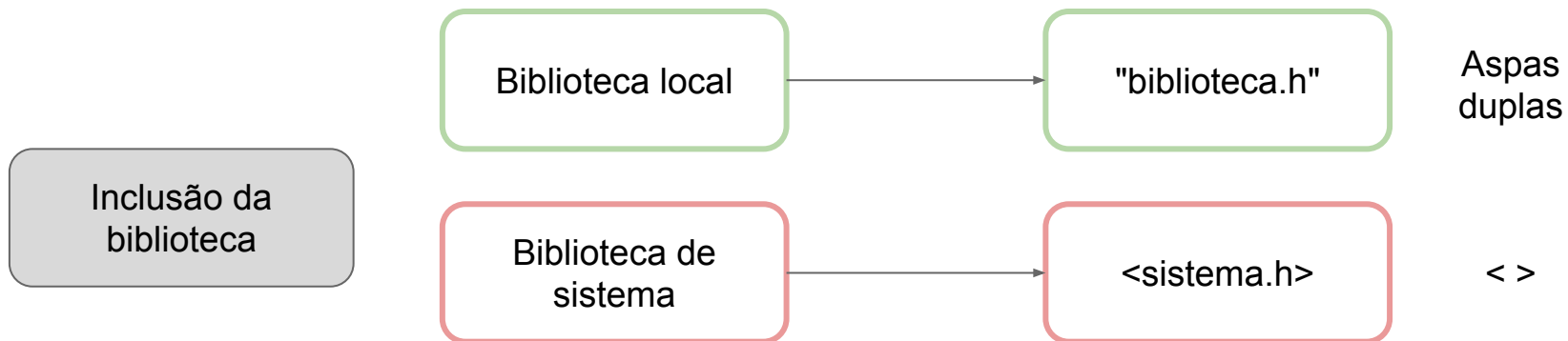
```
#include "calcula.h"

int soma(int a, int b) {
    return a + b;
}
```

```
#include <stdio.h>
#include "calcul.h"

int main() {
    printf("%d\n", soma(1, 4));
    return 0;
}
```

programa.c



```
#include <stdio.h>
#include "calculculo.h"

int main() {
    printf("%d\n", soma(1, 4));
    return 0;
}
```

programa.c

Como compila?


```
gcc -o prog -std=c99 -W -Wall -Wshadow programa.c calculo.c -I.
```

```
gcc -o prog -std=c99 -W -Wall -Wshadow programa.c calculo.c -I.
```

Todos os seus .c

Os .h não

```
gcc -o prog -std=c99 -W -Wall -Wshadow programa.c calculo.c -I.
```



Indica onde estão os .h

**Flag para
inclusão de
bibliotecas**

-I .

**Diretório onde
estão as
bibliotecas**

. representa o
diretório atual

**Testem: Crie uma
biblioteca com as
funções básicas da
matemática**

#endif

