

AULA 7 - Implementação da programação modular

Iniciais dos alunos: *AVC, JPP, PC*

Disciplina: *Programação Modular (INF1301)* – Professor: *Flavio Bevilacqua*

:

1. Espaço de dados

Áreas de armazenamento:

- Possuem nome (um ou mais).
- Possuem tamanho.
- Alocados em um meio.

Exemplos:

$A[j] \Rightarrow$ j-ésima posição do vetor A.

$*ptAux \Rightarrow$ espaço referenciado por ptAux.

$ptAux \Rightarrow$ espaço que contém um endereço de memória.

$ptElemTabSimb \ *ObterElemTabSimb(char \ *ptSimbolo)$

$(*ObterElemTabSimb(char \ *ptSimb)).Id \Rightarrow$ campo Id da estrutura apontada pelo ponteiro retornado pela função.

$ObterElemTabSimb(char \ *ptSimb) \rightarrow Id \Rightarrow$ o mesmo que o anterior.

2. Tipos de dados

define:

- Organização
- Codificação

Data: AAAAMMDD CPF: xxxxxxxxx-xx

- tamanho
- Conjunto de valores permitidos

OBS: Tipos de tipo

- Computacional. Ex: int, char*, float
- Tipos abstratos de dados. Ex: estrutura encapsulada que só é conhecida pelos clientes através de funções de acesso.
- Tipos básicos de usuário (typedef, enum, union, struct).

typedef:

Exemplo:

```
typedef float tpVeloc;
```

```
typedef float tpTempo;
```

```
tpVeloc veloc;
```

```
tpTempo tempo;
```

3. Declaração e definição de elementos

Declarar é vincular a um tipo. Typecast não é declaração.

Definir é alocar um espaço na memória e vincular um espaço a um nome (binding).

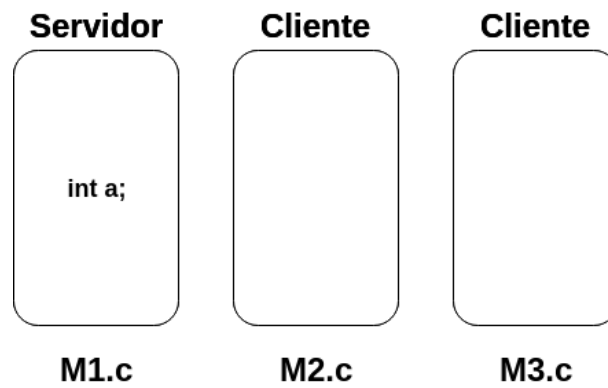
Exemplo:

```
int a  $\implies$  declara e define
```

OBS: Declaramos e definimos ao mesmo tempo quando usamos o tipo computacional. Declaramos sem definir quando declaramos uma struct por exemplo. Definimos sem declarar quando usamos malloc.

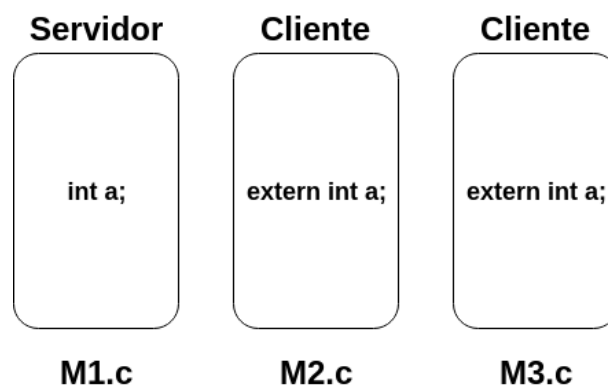
4. Implementação em C e C++

- (a) Declarações e definições de nomes globais exportados pelo módulo servidor.



Ex: `int a;`

- (b) Declarações externas contidas no módulo cliente e que somente declaram o nome sem associá-lo a um espaço de dados.



Ex: `extern int a;`

- (c) Declarações e definições de nomes globais encapsulados no módulo.

Ex: `static int a;`