

Definição de tipos

A palavra reservada `typedef` permite definir (ou redefinir) um tipo de dado. Ela pode ser aplicada a qualquer tipo da linguagem C, mas é particularmente útil com *structs*, pois simplifica a declaração de variáveis e parâmetros de tipo *struct*.

Forma geral:

```
typedef <tipo existente> <novo tipo> ;
```

Exemplo com escalares:

```
typedef unsigned long int uint32_t ; // "uint32_t" equivale a "unsigned long int"

uint32_t a, b ; // aloca duas variáveis do tipo uint32_t
```

Exemplo com *structs*:

```
struct pac_t
{
    char nome[100] ;
    short idade ;
    short quarto ;
} ;

typedef struct pac_t paciente_t ; // struct pac_t -> paciente_t

// declaração e alocação
paciente_t pac1, pac2 ;
```

Ou redefinindo o próprio tipo `paciente_t` ;

```
struct paciente_t
{
    char nome[100] ;
    short idade ;
    short quarto ;
} ;

typedef struct paciente_t paciente_t ;

// declaração e alocação
paciente_t pac1, pac2 ;
```

Ou ainda mais “enxuto”:

```
typedef struct paciente_t
{
    char nome[100] ;
    short idade ;
    short quarto ;
} paciente_t ;
```

```
// declaração/alocação  
paciente_t pac1, pac2 ;
```



O uso de typedef com structs é **desaconselhado** por alguns guias de estilo de código em C, por “esconder” o fato que um determinado tipo é uma struct, o que pode tornar mais difícil a leitura do código.

From:

<https://wiki.inf.ufpr.br/maziero/> - **Prof. Carlos Maziero**

Permanent link:

https://wiki.inf.ufpr.br/maziero/doku.php?id=c:definicao_de_tipos

Last update: **2023/08/15 14:24**

