



MEMÓRIA

Alocação de memória

Estática

```
static int var1;
```

Memória é alocada quando programa é iniciado; variáveis globais, variáveis `static`.

Alocação de memória

Estática

```
static int var1;
```

Memória é alocada quando programa é iniciado; variáveis globais, variáveis `static`.

Automática

```
{  
  char nome[50];  
  int nCal;  
}
```

Memória automaticamente alocada dentro de um bloco e libertada quando o bloco termina.

Alocação de memória

Estática

```
static int var1;
```

Memória é alocada quando programa é iniciado; variáveis globais, variáveis `static`.

Automática

```
{  
  char nome[50];  
  int nCal;  
}
```

Memória automaticamente alocada dentro de um bloco e libertada quando o bloco termina.

Dinâmica

```
int* func() {  
  int* mem = malloc(1024);  
  return mem;  
}  
int* mem = func(); // continua acessível  
free(mem);
```

Memória alocada manualmente; controla-se tamanho, quando é alocada/libertada. **Tem de ser libertada manualmente!**

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

```
void *malloc(size_t size);
```

Aloca um bloco de memória de tamanho `size`. Devolve um apontador para bloco alocado.

```
void *calloc(size_t nmemb,  
             size_t size);
```

Aloca um bloco de memória com `nmemb` elementos de tamanho `size`. Devolve um apontador para bloco alocado.

```
void free(void *ptr);
```

Liberta bloco de memória apontado pelo apontador `ptr`.

```
sizeof( tipo )
```

```
sizeof( char ) -> 1
```

```
sizeof( int ) -> 4
```

```
sizeof( double ) -> 8
```

Devolve tamanho de um tipo.

Tamanhos de tipos podem mudar de sistema para sistema, por isso é mais seguro usar a função `sizeof` em alocações de memória, i.e.



```
calloc(200, sizeof(int));
```



```
calloc(200, 4);
```

```
void * ptr
```

é um apontador genérico para uma zona
de memória

endereço 1 byte de cada vez (à
semelhança do `char`)

Refaça o exercício da cifra de César.

Desta vez peça ao utilizador qual o tamanho máximo da mensagem a receber e faça a alocação dessa memória.

A mensagem recebida é gravada na memória alocada.

Liberte a memória alocada quando o utilizador escolher sair do programa.

Ajuda:

```
char *msg;  
int tamanho_maximo = 5000; // recebido do utilizador  
msg = calloc(tamanho_maximo, sizeof(char)); // alocação  
free(msg); // quando utilizador escolhe sair
```