UNIVERSIDADE FUMEC

Estrutura de Dados I

Alocação Dinâmica de Memória

Profª Amanda Danielle Lima de Oliveira Tameirão

Objetivos

- Compreender o conceito de alocação
- Funções utilizadas
- Alocação para vetores e matrizes

UNIVERSIDADE FUMEC

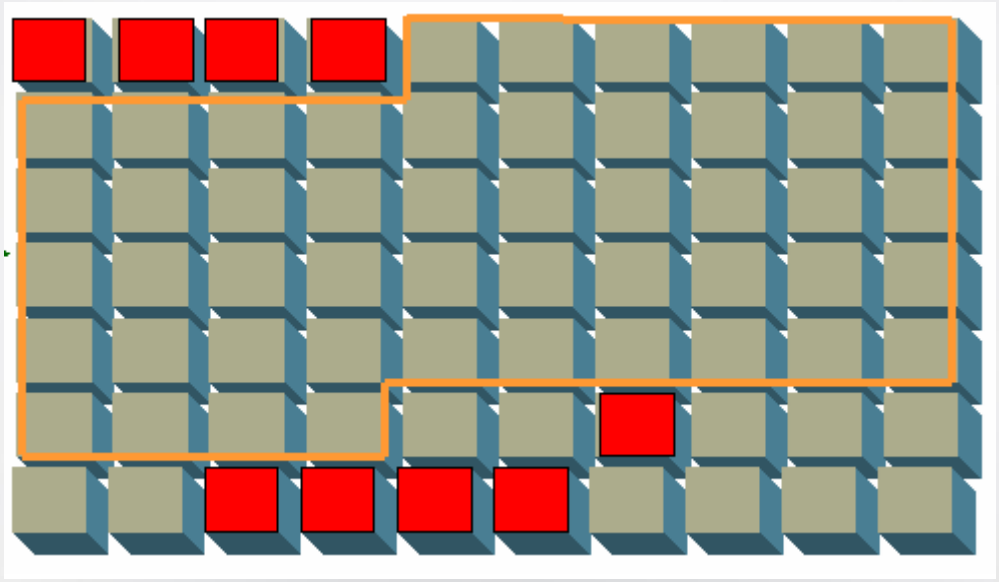
UNIVERSIDADE FUMEC

Alocação Dinâmica

- Definição
- Comandos
- Vetores
- Matrizes

UNIVERSIDADE FUMEC

Definição
Alocação Dinâmica



Alocação Dinâmica

- Permite ao programador alocar espaço de memória em tempo de execução.

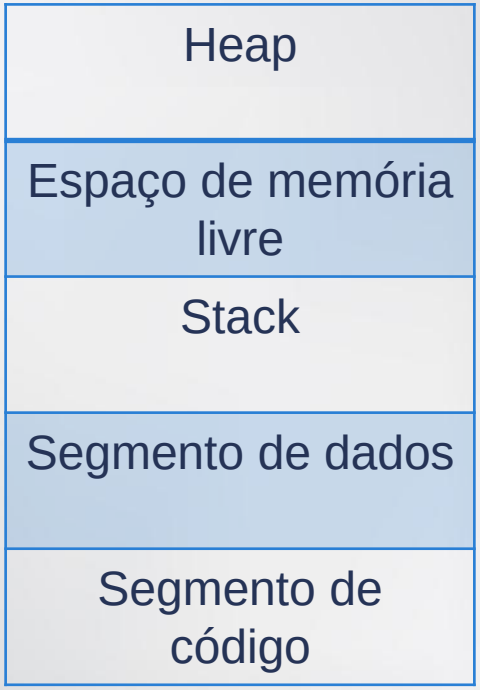
- Existem três formas de definir espaços de memória:

1. Variável global
2. Variável local
3. Alocação dinâmica

Alocação Dinâmica
Gerenciar memória

A memória é divida da seguinte forma

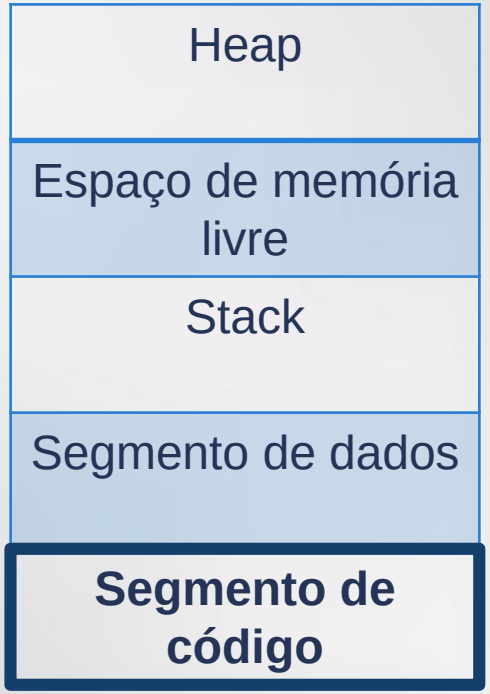
- Segmento de código
- Segmento de dados
- *Stack*
- *Heap*



Alocação Dinâmica
Segmento de Código

Espaço reservado para a execução do código (instruções) do programa.

Geralmente somente leitura



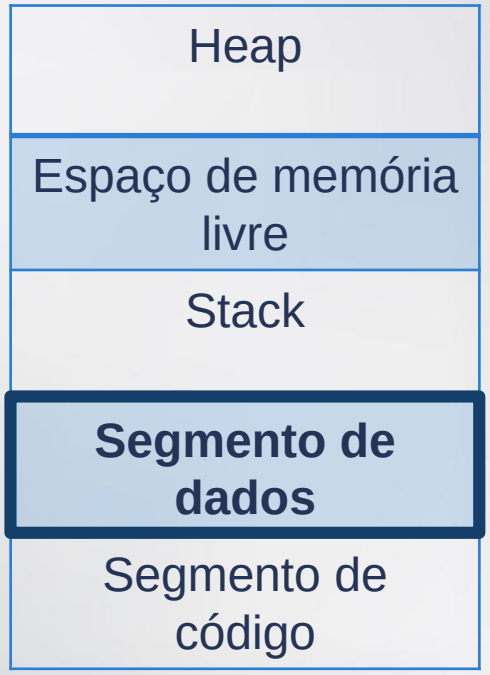
Alocação Dinâmica
Segmento de Dados

Armazena as variáveis globais e estáticas

Tamanho calculado de acordo com as variáveis

Acesso leitura/escrita

Pode ser alterado durante a execução do programa

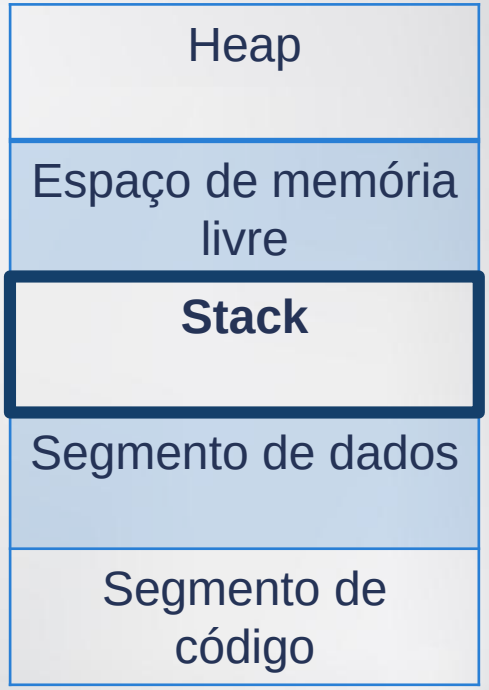


Alocação Dinâmica
Stack

Armazena as variáveis locais e chamadas de funções.

Tamanho variável (depende do SO e compilador)

Ultrapassar o tamanho gera erro de stack – “Stack buffer overflow”



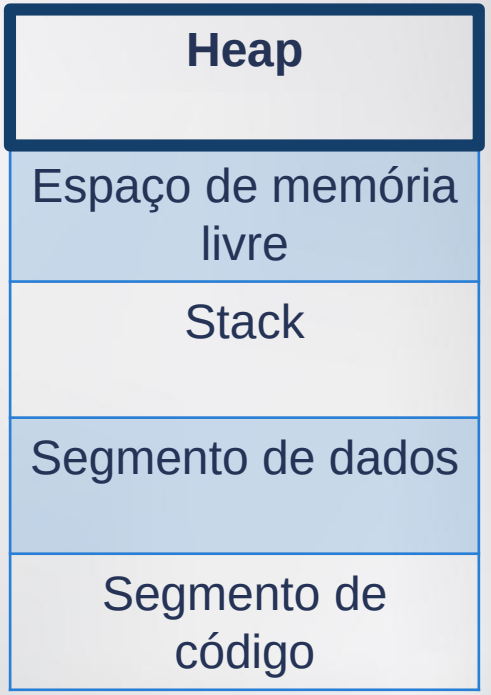
Alocação Dinâmica

Heap

Reservado para alocação dinâmica de memória

Memória reservada pode ser liberada a qualquer momento.

Possui funções específicas para este controle



UNIVERSIDADE FUMEC

Controle de *Heap*

Funções

Controle do *Heap*
malloc()

Função da stdlib.h

Responsável por alocar espaço de memória na *heap*

Necessita de conversão.

O valor retornado é um ponteiro para o início da área alocada. Se ocorrer algum erro, retornará null.

Controle do *Heap*
malloc()

Sintaxe:

```
void *malloc(int tamanho);
```

1. Tamanho é a quantidade de bytes que será alocada.

Dica: Utilize a função sizeof para calcular a quantidade de bytes de um tipo.

Controle do *Heap*
malloc()

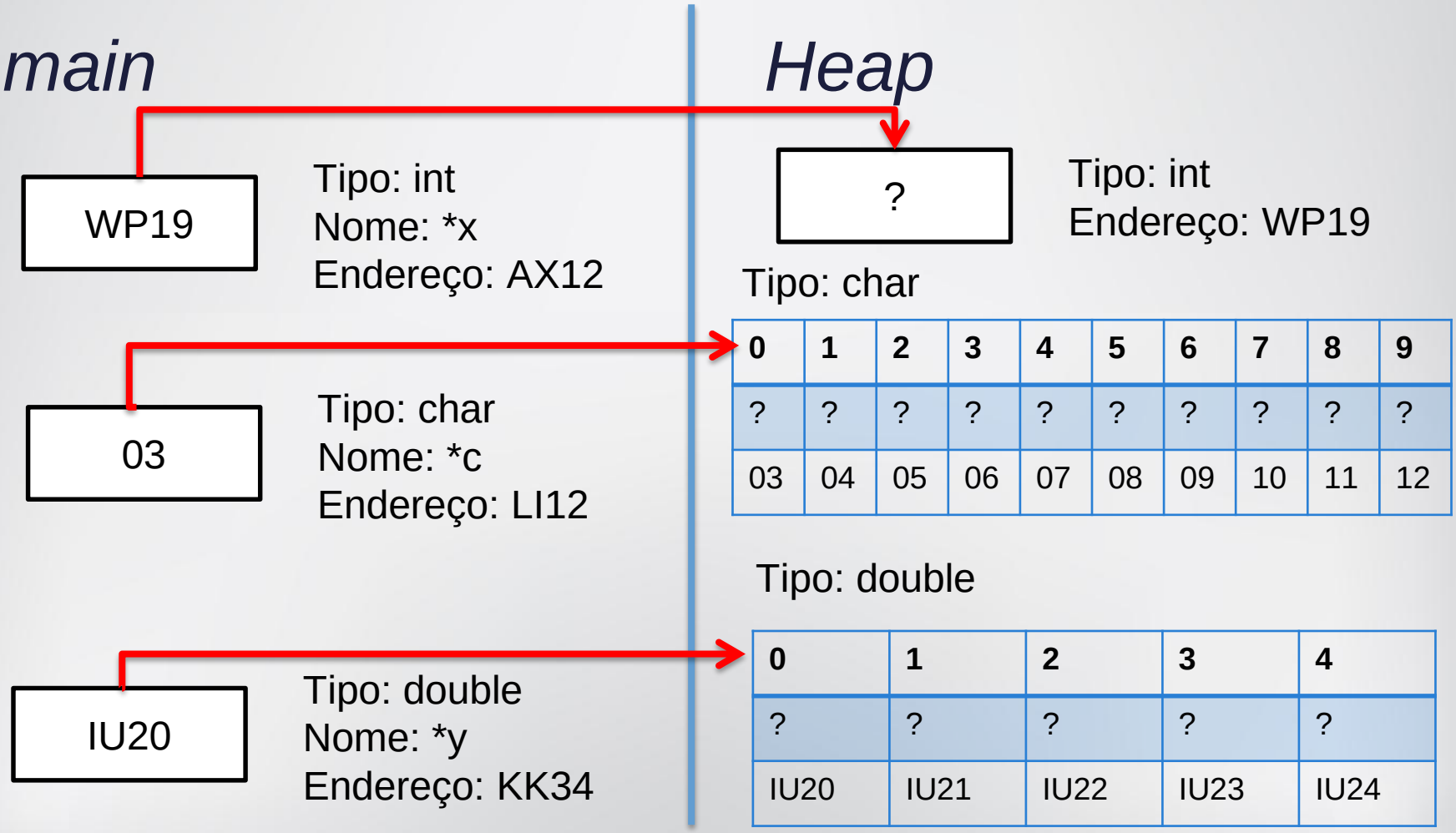
```
#include <stdlib.h>
int main() {
    /*Alocação de um inteiro*/
    int *x;
    x = (int *) malloc (sizeof(int));

    /*Alocação de um vetor de caracteres*/
    char *c;
    c = (char *) malloc (10);

    /*Alocação de um vetor de 5 inteiros*/
    int *y;
    y = (int *) malloc (5 * sizeof(int));
    return 0;
}
```

sizeof – retorna o tamanho, em bytes, de uma estrutura qualquer (variável ou tipo) .

Representação



Controle do *Heap*
free()

Função da stdlib.h

Responsável por liberar o espaço de memória na *heap*

Boa prática da programação

Controle do *Heap*
free()

Sintaxe:

```
void free(void *ponteiro);
```

1. Ponteiro corresponde à variável que efetua a alocação.

Controle do *Heap*
free()

```
#include <stdlib.h>
int main() {
    /*Alocação de um inteiro*/
    int *x;
    x = (int *) malloc (sizeof(int));

    /*Alocação de um vetor de caracteres*/
    char *c;
    c = (char *) malloc (10);

    /*Libera espaço de memória reservado*/
    free(x);
    free(c);
    return 0;
}
```

Representação

main

WP19

Tipo: int
Nome: *x
Endereço: AX12

03

Tipo: char
Nome: *c
Endereço: LI12

IU20

Tipo: double
Nome: *y
Endereço: KK34

Heap

Controle do *Heap*
calloc()

Função da stdlib.h

Responsável por armazenar, em blocos, espaço de memória na *heap.* Já inicializa este espaço com zero.

Controle do *Heap*
calloc()

Sintaxe:

void *calloc(int quantidade, int tamanho);

1. Quantidade corresponde à quantidade de espaços que serão reservados do tamanho indicado.
2. Tamanho é a quantidade de bytes que será alocada.

Controle do *Heap*

calloc()

```
#include <stdlib.h>

int main() {
    /*Alocação de um inteiro*/
    int *px;
    px = (int *) calloc (30, sizeof(int));


    /*Libera espaço de memória reservado*/
    free(px);
    return 0;
}
```

UNIVERSIDADE FUMEC

Exercícios

Lista de exercícios de alocação e ponteiros

Bibliografia

CELES, Waldemar et al. Introdução a estrutura de dados: com técnicas de programação em C. Rio de Janeiro: Elsevier, 2004.

Curso de Linguagem C – UFMG. Disponível em <http://www.inf.ufsc.br/~fernando/ine5412/C_UFMG.pdf> Acesso em 22 de abril de 2014

Notas de aula do Prof. Flavio Lapper

Notas de aula do Prof. Rafael Nunes