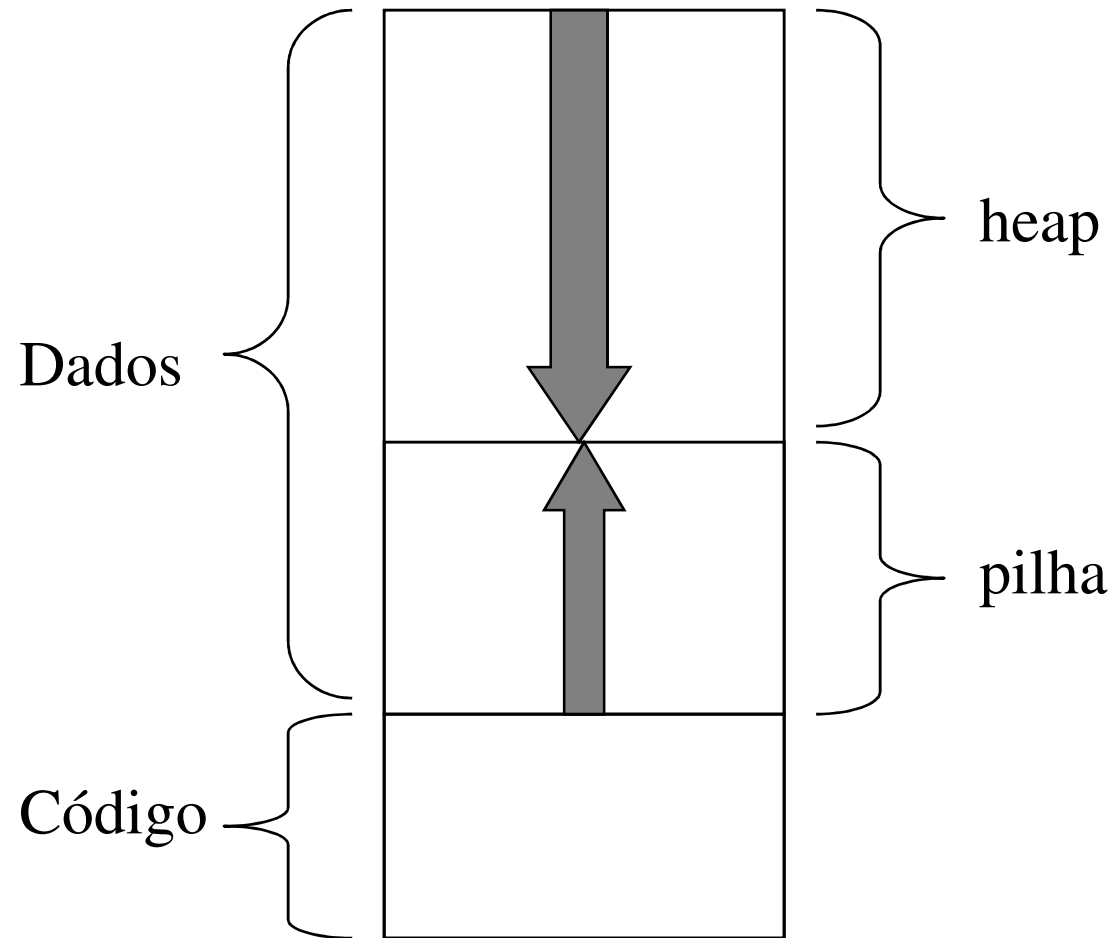


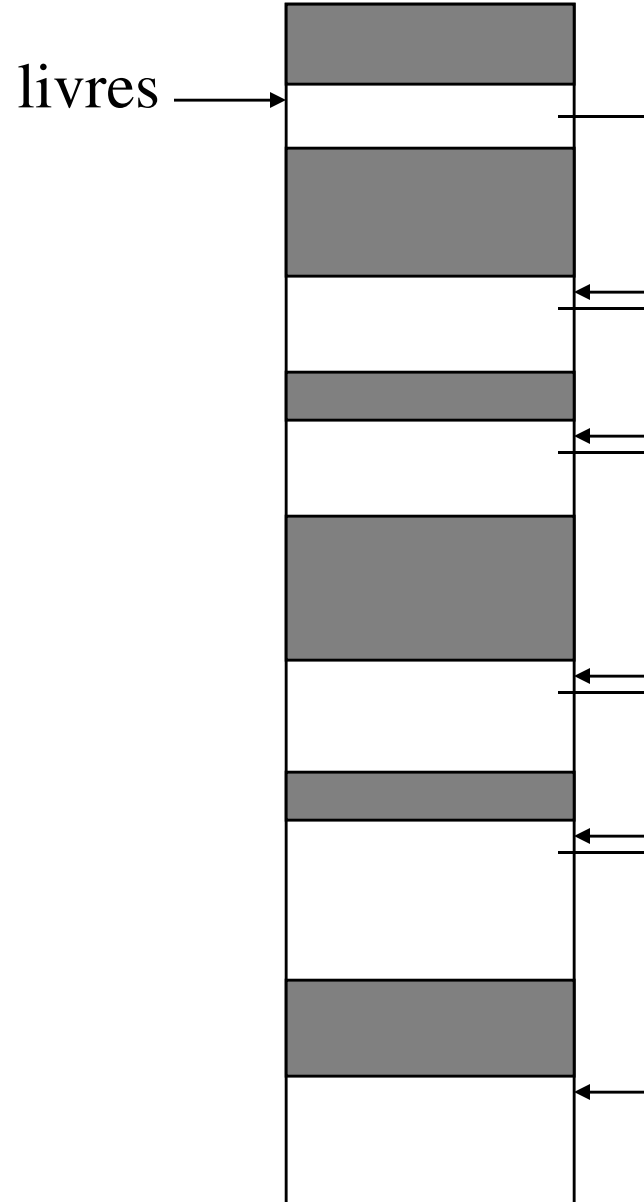
Alocação dinâmica de memória

- As funções **malloc**, **calloc** e **free** são responsáveis pela alocação dinâmica de blocos de memória (análogas a **new** e **dispose** do Pascal).
- A alocação é feita dentro do espaço de endereçamento reservado ao programa pelo sistema operacional, antes do início da execução.

Organização da memória



Heap



Alocação

- Como atender a um pedido de alocação de memória como `malloc(n)` ?
- Estratégias: percorrer a lista de blocos livres e alocar um pedaço
 - do primeiro bloco que atende ao pedido (*first fit*)
 - do bloco que melhor atende ao pedido e deixa o menor resíduo (*best fit*)

Liberação de memória

- Ao liberar um bloco de memória, se o mesmo for contíguo a um bloco livre (ou dois) estes devem ser unidos num único bloco.
- problema: como detectar a vizinhança ?
 - estratégias:
 - lista de blocos livres ordenada pelos endereços
 - 'marcas de fronteira' em cada bloco.

Fragmentação

- Após um certo número operações de alocação e liberação de memória, a memória pode chegar a uma situação tal que
 - o total de memória livre é maior que um certo n
 - um pedido de alocação de um bloco de tamanho n não pode ser atendido porque não existe um bloco contíguo com tamanho maior ou igual a n .

Estratégias para evitar (adiar) a fragmentação

- Alocar os blocos como múltiplos de um certo n (p. ex. $n = 256$)
- Rearranjar a memória: só é possível em linguagens como Java e C#, que mantêm em tempo de execução a descrição dos dados definidos pelo programa.

Garbage Collection

- Em linguagens como Lisp, Java e C# o programador não precisa se preocupar em liberar os blocos de memória que já não são utilizados pelo programa.
- A recuperação dos blocos não utilizados é feita automaticamente pela função de 'coleta de lixo' (garbage collection).

Garbage Collection

- Duas abordagens:
 - *contador de referências*: cada bloco mantém um contador indicando quantos apontadores do programa fazem referência ao mesmo. Se esse contador chega a zero, o bloco é liberado.
 - *marcação*: periodicamente a memória é varrida e os blocos acessíveis pelo programa são marcados. Ao final da varredura, os blocos não marcados são liberados.

Garbage Collection

- Problemas:
 - *contador de referências*: falha no caso se referências circulares (mesmo assim é usado no Windows 95, 98, NT e 2000).
 - *marcação*: leva um tempo proporcional ao número de blocos alocados.