

# Exercise classes computer architecture and system software

Yves Younan

DistriNet, Department of Computer Science

Katholieke Universiteit Leuven

Belgium

[Yves.Younan@cs.kuleuven.ac.be](mailto:Yves.Younan@cs.kuleuven.ac.be)

# Overview

- Structures
- Linked lists
- Heap

# Structures

- Also called records
- Like objects but no member functions, only datamembers
- Example in C:

```
struct complex {  
    int r;  
    int i;  
}
```

- Stored next to each other in memory
- Members can be different types, aligning on 4 bytes

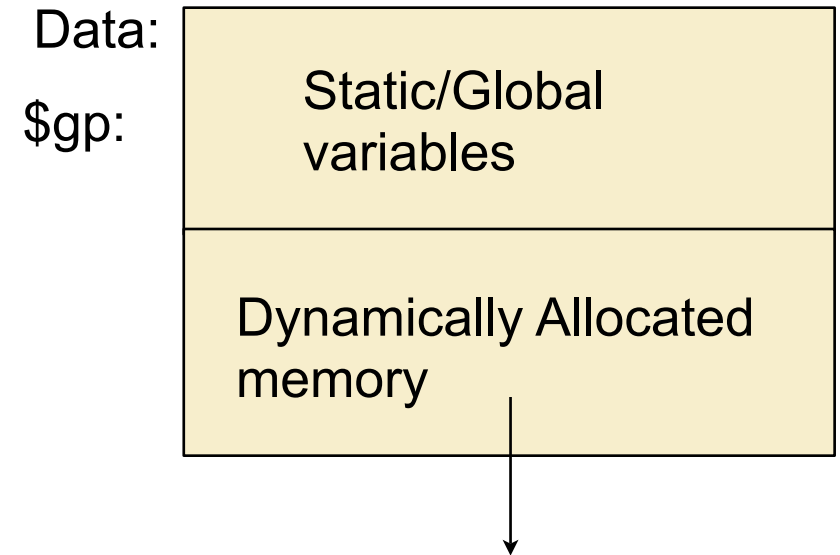
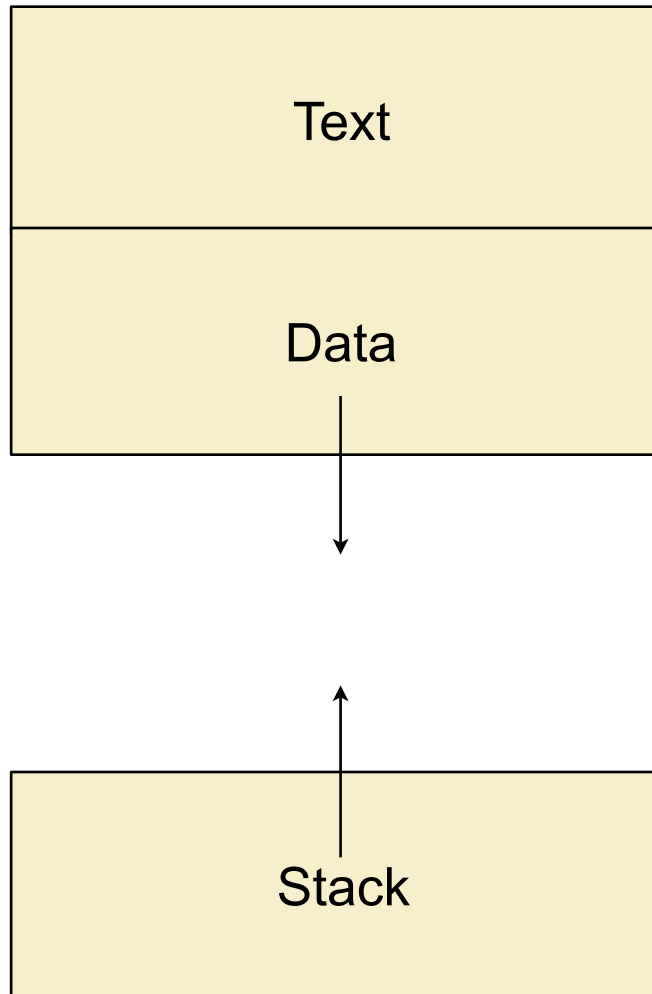
# Structures in memory

- Pointer to the top of the structure, accessed by offset from the top:
  - struct complex;
  - `complex->a = complex+0`
  - `complex->b = complex+4`

# Linked lists

- Linked lists are structures that contain a pointer to the next element
  - struct element {
    - struct element \*next;
    - int key;
    - char value[100];
  - }
  - element->next is a pointer to the next element, element->next->next is a pointer to the element after the next element.

# Program memory layout



# Heap

- Contains dynamically allocated memory: memory which is only allocated at run time
  - Example: array of which the size is not known
- Usually allocated using the malloc/new library call which is a wrapper for the brk system call
- Assume that heap starts at \$gp for the exercises and that you don't need to increase it with brk