The C Language

CS238P: Principles of operating systems - Fall’18

Aftab Hussain
(Adapted from Vikram Narayanan’s slides for ICS143A Fall’17)

University of California, Irvine
Data and Computation
Data can be of different types.

- char (1 byte)
- int, long (4/8 bytes)
- pointer (1 byte), structs, etc.

They can also be:

- constants
- variables
A data type therefore determines two things\(^1\):

- the size of the data variable
- how the data is to be interpreted.

\(^1\)https://www.tutorialspoint.com/cprogramming/c_data_types.htm
Computation
Statements

- declarations
- assignments
- for, do...while, while
Hw1(xv6 shell)

- if...else
  
  ```c
  pid = fork();
  if(pid == -1)
    perror("fork: ");
  ```
• **if...else**
  
  ```c
  pid = fork();
  if (pid == -1)
    perror("fork:");
  ```

• **switch...case**

  ```c
  switch(cmd->type){
  case '>'; ...; break;
  default: ...; break;
  }
  ```
Hw1(xv6 shell)

- if...else
  
  ```
  pid = fork();
  if(pid == -1)
      perror("fork:");
  ```

- switch...case
  
  ```
  switch(cmd->type){
  case '>'; ...; break;
  default: ...; break;
  }
  ```

- Functions
  
  - Process creation (fork, exec)
  - File I/O (open, close, read, write)
Pointers

p points to integer a by storing a's address. 1 byte is used. (not showing how the address is actually stored here in binary)

Integer a stored in the memory containing the decimal value 5. 4 bytes are used.

(a)

```
int a = 5;
int *p = &a;
```

(b)

Fig. 1(a). Simple illustration of how a pointer points to data in the memory.
(b) Corresponding C code for Fig. 1(a).
• Collection of objects of the same data type
Arrays

- Collection of objects of the same data type
- Accessed by index (0 ... size - 1)
• Collection of objects of the same data type
• Accessed by index (0 ... size - 1)
• String is an array of characters
Designated Initializers

```c
#define CAPSLOCK (1<<3)
#define NUMLOCK (1<<4)
#define SCROLLLOCK (1<<5)

static uchar togglecode[256] = {
    [0x3A] CAPSLOCK,
    [0x45] NUMLOCK,
    [0x46] SCROLLLOCK
};
/* equivalent to */
togglecode[0x3A] = CAPSLOCK;
togglecode[0x45] = NUMLOCK;
togglecode[0x46] = SCROLLLOCK;
```

Initialize the array elements 0x3A, 0x45, 0x46 only

---

2 http://gcc.gnu.org/onlinedocs/gcc-4.0.4/gcc/Designated-Ins.html
3 sheet 77, xv6-rev9.pdf
Examples

(arrays-ptrs.c & arrays-strings.c)