Recitation 0

TA Hanxiong Chen
hc691@rutgers.edu
Office hour: Thu 3:00-4:00pm, Hill 206

Questions

- How to learn C??

  Learn by using it!

- Where to ask for help?

  man(in Linux); Google (stackoverflow); Classmates; Professor & TAs

What is the key?

- The key of C is POINTER!
- To some degree, to understand pointer is to understand memory management.

Use iLab

- Register: https://www.cs.rutgers.edu/resources/instructional-lab
- Choose a server: http://report.rutgers.edu/mrtg/ilab.html
- Use command line to do ssh login
  - Linux/UNIX, MacOS (terminal)
  - Windows use PuTTY
- Example:
  - ssh netid@command.cs.rutgers.edu
  - Then you are asked to type in your password (the same as your password to login sakai)
  - If you want to use ssh key instead of password, please google “ssh-keygen” for some tutorials.

Supplements (in C language)

- You must use constant value to define an array.
  - int a[10];
  - Or use macro:

```
#define NUM 10
…
int a[NUM];        // the same as int a[10]
```

- String is a char array in C (different from java), which is end with “\0”.
  - Question: char s[?] = “Hello”;
  - What is “?” ? -- A. 12 B.13 C.14 D.15
  - *char s[6] = “Hello”;* the same as *char s[6] = {‘H’, ‘e’, ‘l’, ‘l’, ‘o’, ‘\0’};*

Supplements (in C language continue...)

- Use malloc to do dynamic memory allocation.
  - Malloc return void*. C doesn't do boundary check.
- I recommend you to do initialization once you define a new pointer variable.
  - char *s = NULL;
  - Or char *s = (char*)malloc(10*sizeof(char));
  - Uninitialized pointer is called wild pointer. (It points to unexpected place. It's dangerous!)
- You MUST FREE the memory which you malloc manually!! (different from java)
- Set NULL to the freed pointer!!
  - A dereferenced pointer which is not pointed to NULL is called dangling pointer.

Supplements (gcc/gdb)

- To compile
  - gcc source_file_dir -o result_file_name -g
  - Try to use gcc --help to get more info of other choices (or use man)
- To debug
  - gdb executable_file_dir
  - Enter debug mode:
    - you can use “b line_number” to set breakpoint. E.g. “b 30” stands for set breakpoint at line 30.
    - “r” command to run the program. You can add arguments after that.
    - “s” command to run step-by-step. (you can step into a function by using s)
    - “c” command to run to your next breakpoint
    - “q” to quit gdb
    - “p” to print the value of a variable

Others

- Pointer
  - To define a pointer: int *p;
  - To get the content of a pointer: *p

E.g. int a = 3;

```
int * b = &a;          //define a int pointer b and initialize it with the address of variable a

printf("%d\n", *b);    //dereference pointer b and print it out
```

Result: 3

Other (continue...)

- asprintf()/snprintf() vs sprintf()
  - sprintf() is not safe. May result in buffer overflow
  - E.g. (1): char* x = (char*) malloc(2 * sizeof(char)); sprintf(x,"%s%s", "12", "34");
  - E.g. (2): char *x = (char *) malloc(2 * sizeof(char)); int size = snprintf(x, 2, "%s%s", "12", "34");
  - E.g. (3): char *x; int size = asprintf(&x, "%s%s%s", "12", "34", "56");
  - REMEMBER TO FREE THE POINTER!!

The same thing goes with “gets()” vs “fgets()”

Other (continue...)

- The same thing goes with “gets()” vs “fgets()”
  - gets() is not safe
  - fgets(char*, length, FILE *fp)

Others (cont’d)

1. Write a program to print this triangle:

```
*
**
***
****
*****
******
*******
********
*********
**********
```

Thank you!

TA Hanxiong Chen
hc691@rutgers.edu
Office hour: Thu 3:00-4:00pm, Hill 206