Review of C

Program Design (II)

2022 Spring

Fu-Yin Cherng
Dept. CSIE, National Chung Cheng University

Review of C

- Basic structure of a simple C program
- Variables and Types
- Assignment
- Formatted Input/Output
- Expression
- Selection Statements
- Loops
- Basic Types
- Arrays
- Functions
- Pointers

Instead of directly explaining each topic,
I will raise several questions that you should be able to answer
in your head to help you review.

Basic structure of a simple C program
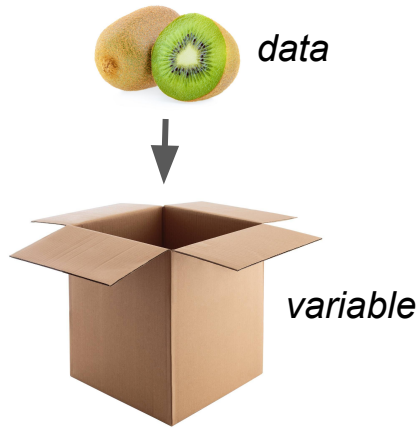
- Can you identify where is directive, function, and statement?
- What is the output of this program?

```c
#include <stdio.h>

int main(void)
{
  printf("Hello CCU.\n");
  return 0;
}
```

Variables and Types

- How to store data and value in C?
- Every variable must have a **type,** which specifics what kind of data it will store
- What is the difference between data with type of `int` and `float`

data

variable

Assignment

- How to store data in the variables?
- We give a value to a variable by assignment (=)
- What happen if we assign the incorrect type of data to the variable

```
int height, length, width, volume;
float profit, loss;

height = 8;
length = 12;
width = 10;
profit=2150.48;
height = 8.1
```

Formatted Input/Output

- How to print value of variable?
- How to read input from users?
- What is "`%d`"?
- If we use "`%4d`" with `printf`, what will be the printed message?
- What is the meaning of `&` ?

```
int x = 1;
printf("%d", x);
scanf("%d", &x);
```

Expression

- Operators
  - arithmetic, assignment, relational, logial, …
- Arithmetic Operators
  - Operator Precedence and Associativity
- Assignment Operators
  - Simple and Compound assignment

Arithmetic and Assignment

Can you remember the effects of these symbols on variable?
What is the meaning of precedence and associativity?

Precedence	Name	Symbol	Associativity
1 (high)	increment (postfix)	`variable++`	left
	decrement (postfix)	`variable--`	
2	increment (prefix)	`++variable`	right
	decrement (prefix)	`--variable`	
	unary plus	`+variable`	
	unary minus	`-variable`	
3	multiplicative	`* / %`	left
4	additive	`+ -`	left
5 (low)	(compound) assignment	`= *= /= %= += -=`	right

Selection Statements (1/2)

- Do you remember how to use Relational, Equality, and Logical Operators?
- What is the difference between these operators?
- How to use them with arithmetic and assignment operators?

Symbol	Meaning
<	less than
>	greater than
<=	less than or equal to
>=	greater than or equal to

Relational Operator

Symbol	Meaning
==	equal to
!=	not equal to

Logical Operator

Selection Statements (2/2)

Can you think a simple example of how to use these statements?

- `if()` and `else()` statement
- `switch()` statement
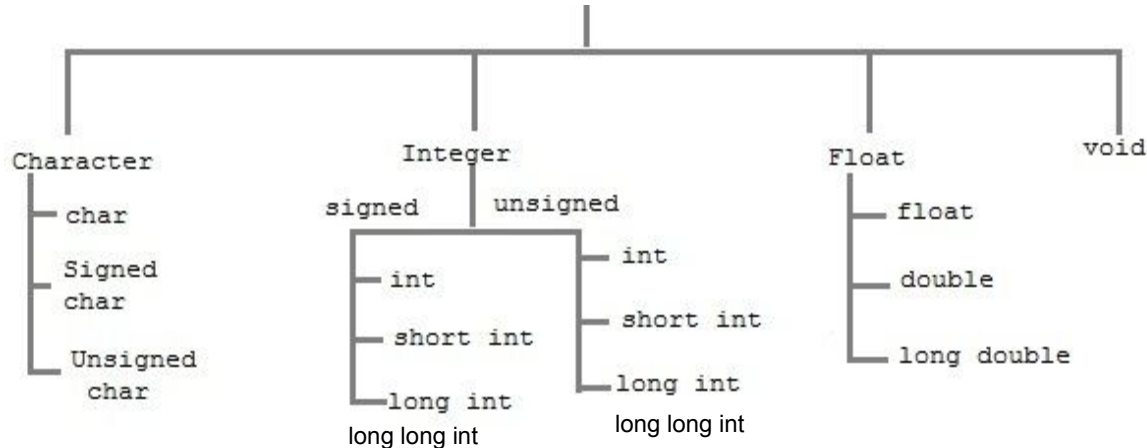- conditional expression: `expr1 ? expr2 : expr3`

Loops

- What is loops?
- What is the difference between `while` and for `loop`?
- How to use `break` and `continue` statements with loops?

```
while ( expression ) statement

for ( expr1 ; expr2 ; expr3 ) statement
```
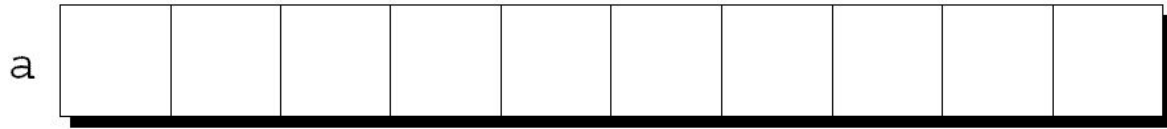
Basic Types

- What is the difference between these types?
- What is *ASCII*?
- Can you make an example of how to use `getchar()` and `putchar()`?
- How to convert variable of one type to another?
- How to define a new type by using `typedef`?

```
                                    |
      ┌─────────────────────────────┼─────────────────────────┐
      │                             │                          │
Character                        Integer                     Float                  void

  ├─ char           signed    │    unsigned              ├─ float
  │                           │      ┌─ int
  ├─ Signed           ├─ int  │      │                   ├─ double
  │   char            │       │      ├─ short int
  │                   ├─ short int   │                   └─ long double
  └─ Unsigned         │              └─ long int
      char            └─ long int
                    long long int    long long int
```

Arrays

- What is array? How many dimensions do the `a` array have?
- How to declare and initialize the array `a`?
- How to access each element inside `a`? What is array indexing?
- What is Multidimensional Array and how to create one?

*The elements of a one-dimensional **array a** are **conceptually** arranged one after another in a single row (or column):*
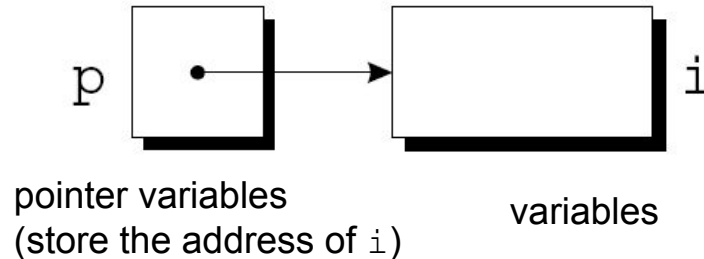
Functions

- What is function? What is the advantage of using function?
- How to Defining and Calling Functions?
- What is difference between Arguments and Parameters?
- What is the difference between global and local variables?

*return-type function-name (**parameters**)*

{

   declarations or statements

}

Pointers

Super important! Please review this topic if you can't answer these questions!

- What do we mean when we said *the address of a variable*?
- Why do we need pointer variable?
- How to declar pointer variables?
- What is the effect of Address (`&`) and Indirection (`*`) Operators?
- How to use pointer with Multidimensional array and function?
- ….

pointer variables
(store the address of `i`)

variables

Let's all for the review of C