CS 50 - 1

Introduction

We can convert **source code** into **machine code** using a very special piece of software called **compiler**.

Code can be evaluated upon three axes.

- **Correctness**; refers to "does the code run as intended?"
- **Style**; refers to "how aesthetically pleasing and consistent is the code?"
- **Design**; refers to "how well is the code designed?"

VSCode

Visual studio code is a compiler with a lot of softwares pre- loaded on it.

We can build a program in C by typing `~$ code file-name.c` into the **terminal** window. Then, in the **text editor** that appears, you can write your cod

```
#include <stdio.h>
int main(void)
{
    printf(« hello, world\n »);
}
```

Clicking back in the terminal window, you can compile your code by executing `make filename`, Then type `./file_name` and your program will execute.

Functions

In C, to display any text on the screen, there is a function called printf.

```
printf("hello, world");
```

A common error in C programming is the omission of a semi-colon.

The argument passed to printf is "hello, world".

Variables

```
# include <stdio.h>

int main(void)
{
    string answer = get_string("what's your name ?");
    printf("hello, %s\n", answer);
}
```

answer is a special holding place we call a variable.

There are many data types, such as int, bool, char, etc...

Conditional

```
int main(void)
{
    int x = get_int("x? ");
    int y = get_int("y? ");

    if (x < y)
    {
        printf("x is less than y\n");
    }
}
```

== ensure that something is equal to something else, where a single equal sign would have a very different function in C.

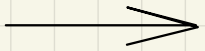
|| means or

&& means and

Loops

Loops give a better design to your code

```
printf("meow\n");
printf("meow\n");
printf("meow\n");
```



```
int i = 0;
while (i < 3)
{
    printf("meow\n");
    i++;
}
```

We can further improve the design using a for loop. for loops include 3 arguments.

```
for (int i = 0; i < 3; i++)
{
    printf("meow\n");
}
```

The first argument starts our counter at zero.

The second argument is the condition that is being checked.

The last argument increments by one each time the loop runs.

Linux and the command line

Linux is an operating system that is accessible via the command line in the terminal window of VSCode.

Some common command-line arguments we may use, include:

- `cd`, for changing the current directory
- `cp`, for copying files and directories
- `ls`, for listing files in a directory
- `mkdir`, for making a directory
- `mv`, for moving files and directories
- `rm`, for removing files
- `rmdir`, for removing directories

Comments

Comments are fundamental parts of computer programs, where you leave explanatory remarks to yourself and others that may be collaborating with you regarding your code.

Typically each comment is a few words or more, providing the reader an opportunity to understand what is happening in a specific block of code. Further, such comments serve as a reminder for you later when you need to revise your code.

```
// print grid of bricks
for (int i=0; i<n; i++)
{
    for (int j=0; j<n; j++)
    {
        printf("#");
    }
    printf("\n");
}
```

Comments involve placing // into your code, followed by a comment.

Abstraction

Abstraction is the art of simplifying our code such that it deals with smaller and smaller problems.

We can abstract away two problems into separate functions.

```
int get_size(void)
{
    ---
}
```

```
void print_grid(int n)
{
    ---
}
```

```
int main(void)
{
    int n = get_size();
    print_grid(n);
}
```

Operators and types

Operators refer to the mathematical operations that are supported by your compiler. In C, these mathematical operators include:

- + for addition
- - For subtraction
- * For multiplication
- / for division
- % for remainder

Types refer to the possible data that can be stored within a variable. Types are very important because each type has specific limits. Because of the limits in memory, the highest value of an int can be 4294967296.

bool	a Boolean expression of either true or false
char	a single character like a or 2
float	a floating-point value, or real number with a decimal value
double	a floating-point value with more digits than a float
int	integers up to a certain size, or number of bits
long	integers with more bits, so they can count higher than an int
string	a string of characters