C++ for C Coders & Data Structures
Lecture Notes by idebtor@gmail.com

Command line processing

The most important function of C/C++ is main() function. It is mostly defined with a return type of int and without parameters :

```
int main() { /* ... */ }
```

We can also give command-line arguments in C and C++. Command-line arguments are given after the name of the program in command-line shell of Operating Systems.
To pass command line arguments, we typically define `main()` with two arguments : first argument is the number of command line arguments and second is list of command-line arguments.

```
int main(int argc, char *argv[]) { /* ... */ }
```

or

```
int main(int argc, char **argv) { /* ... */ }
```

- argc (ARGument Count) is int and stores number of command-line arguments passed by the user including the name of the program. So if we pass a value to a program, value of argc would be 2 (one for argument and one for program name)
- The value of argc should be non negative.
- argv(ARGument Vector) is array of character pointers listing all the arguments.
- If argc is greater than zero, the array elements from argv[0] to argv[argc-1] will contain pointers to strings.

Argv[0] is the name of the program , After that till argv[argc-1] every element is command -line arguments.

```
// Name of program argcargv.cpp
#include <iostream>
using namespace std;

int main(int argc, char** argv) {
    cout << "You entered " << argc << " arguments:" << endl;

    for (int i = 0; i < argc; ++i)
        cout << argv[i] << endl;

    return 0;
}
```

Input:

```
$ g++ argcargv.cpp -o arg
$ ./arg command line args
```

Output:

```
You entered 4 arguments:
./arg
command
line
args
```

Properties of Command Line Arguments:

```markdown
- They are passed to main() function.
- They are parameters/arguments supplied to the program when it is invoked.
- They are used to control program from outside instead of hard coding those values inside the code.
- argv[argc] is a NULL pointer.
- argv[0] holds the name of the program.
- argv[1] points to the first command line argument and argv[n] points last argument.

__Note__: You pass all the command line arguments separated by a space, but if argument itself has a space
then you can pass such arguments by putting them inside double quotes “” or single quotes ”.

### Reference:
https://www.geeksforgeeks.org/command-line-arguments-in-c-cpp/

----------------------------
_One thing I know, I was blind but now I see. John 9:25_
```