

Tutorial VIII

December 12

Exercise 1 [*Basic command line options*] Write a program that takes three parameters from the command line.

- The first one, let us call it `op`, should be an operator of the following list: `+`, `-`, `*` and `/`.
- The second and third positional arguments, let us call them `A` and `B`, must be numbers.

The program should compute `A op B` and output the result. For instance, if the executable is called `simple_clo`, the following input

```
./simple_clo + 45 9.2
```

must produce

```
54.2
```

Note: The symbol `*` has a special meaning for the shell. Pay attention to how you input it.

Exercise 2 [*Inside the unit circle*] Write a program that expects the following type of input

```
n
x1 y1
x2 y2
...
xn yn
```

where `n` is the number of lines that will follow and each of them contains two cartesian coordinates separated by a space.

Determine, **after reading all the input**, how many points are inside the unit circle and print them out.

Hint: Use dynamical allocation of memory after reading the number `n`.

For example, after reading the following input

```
5
0.3 2
0.9 0.02
1 1
47 0
0.5 0.5
```

the program should print out

```
There are 2 points inside the unit circle with coordinates
0.9 0.02
0.5 0.5
```

Excercise 3 [*Organization of code*]

- (i) Write a function with the following signature

```
double add(double a, double b, double c)
```

that adds three numbers.

Call that function from your main function to test it, e.g. `printf("%gn", add(3.0, 5.0, 7.0));`

- (ii) Such a large and complex code deserves that we devote some time to organize it a bit better. Structure your code in the following way. Put your `add` function in a different file, called `mymath.c` and your `main` function in a file called `main.c`. Add appropriate declarations such that the code still compiles and runs properly.
- (iii) Further improve the structure by moving the declarations to a header file, e.g. `mymath.h`, and use suitable `#include` statements in the files of your project such that everything works as it should.