C programming for beginners

Lesson 0

December 9, 2008

Objectives

- The best way to learn C is solving a practical problem.
- We will solve a not so simple problem using a basic C knowledge.

Objectives

- The best way to learn C is solving a practical problem.
- We will solve a not so simple problem using a basic C knowledge.
- Task: Plot the Mandelbrot set.

Main task

- What are the values of c that hold

$$x_{n+1} = {x_n}^2 + c \quad (x, c\,\epsilon\,\mathbb{C})$$

bounded?

Partial task

- Task 0: Print the values of

$$x_{n+1} = {x_n}^2 + c \quad (x, c \epsilon \mathbb{R})$$

for a given c value.

Program

```
#include <stdio.h>
#include <stdlib.h>

double next (double x, double c){
        return x*x+c;
}

int main(int argc, char *argv[]){
        int i,n;
        double x,c;
        c = atof(argv[1]);
        n = atoi(argv[2]);
        x = 0;
        for (i=0;i<n;i++){
                printf(" %g\n", x);
                x = next(x,c);
        }
        return 0;

}
```

Preprocessing directives

```
#include <stdio.h>
#define HUNDRED 100
```

Program

```
#include <stdio.h>
#include <stdlib.h>

double next (double x, double c){
        return x*x+c;
}

int main(int argc, char *argv[]){
        int i,n;
        double x,c;
        c = atof(argv[1]);
        n = atoi(argv[2]);
        x = 0;
        for (i=0;i<n;i++){
                printf(" %g\n", x);
                x = next(x,c);
        }
        return 0;

}
```

Preprocessing directives

```
#include <stdio.h>
#define HUNDRED 100
```

Functions

```
double funcion (int foo, ...){
    double bar;
    bar = sqrt(foo);
    return bar;
}
```

Program

```
#include <stdio.h>
#include <stdlib.h>

double next (double x, double c){
        return x*x+c;
}

int main(int argc, char *argv[]){
        int i,n;
        double x,c;
        c = atof(argv[1]);
        n = atoi(argv[2]);
        x = 0;
        for (i=0;i<n;i++){
                printf(" %g\n", x);
                x = next(x,c);
        }
        return 0;

}
```

Variables

- short, int, long int
- float, double
- char
- void

Program

```
#include <stdio.h>
#include <stdlib.h>

double next (double x, double c){
        return x*x+c;
}

int main(int argc, char *argv[]){
        int i,n;
        double x,c;
        c = atof(argv[1]);
        n = atoi(argv[2]);
        x = 0;
        for (i=0;i<n;i++){
                printf(" %g\n", x);
                x = next(x,c);
        }
        return 0;

}
```

main() syntax

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

Program

```
#include <stdio.h>
#include <stdlib.h>

double next (double x, double c){
        return x*x+c;
}

int main(int argc, char *argv[]){
        int i,n;
        double x,c;
        c = atof(argv[1]);
        n = atoi(argv[2]);
        x = 0;
        for (i=0;i<n;i++){
                printf(" %g\n", x);
                x = next(x,c);
        }
        return 0;

}
```

main() syntax

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

Control structures

```
for (i=0;i<n;i++){
        .....
}
```

Program

```
#include <stdio.h>
#include <stdlib.h>

double next (double x, double c){
        return x*x+c;
}

int main(int argc, char *argv[]){
        int i,n;
        double x,c;
        c = atof(argv[1]);
        n = atoi(argv[2]);
        x = 0;
        for (i=0;i<n;i++){
                printf(" %g\n", x);
                x = next(x,c);
        }
        return 0;

}
```

main() syntax

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

Control structures

```
for (i=0;i<n;i++){
    .....
}
```

printf()

```
printf("format",variables);
```

soto@new-host-2:~/uned/curso C/presentaciones/translation/mandelbrot

```
[soto@new-host-2 mandelbrot]$ ./prog0 -1.5 20
0
-1.5
0.75
-0.9375
-0.621094
-1.11424
-0.258464
-1.4332
0.554053
-1.19303
-0.0766894
-1.49412
0.732391
-0.963604
-0.571468
-1.17342
-0.123075
-1.48485
0.704787
-1.00328
[soto@new-host-2 mandelbrot]$
```

soto@new-host-2:~/uned/curso C/presentaciones/translation/mandelbrot

```
[soto@new-host-2 mandelbrot]$ ./prog0 0.5 20
0
0.5
0.75
1.0625
1.62891
3.15334
10.4435
109.567
12005.5
1.44131e+08
2.07739e+16
4.31554e+32
1.86239e+65
3.46848e+130
1.20304e+261
inf
inf
inf
inf
inf
[soto@new-host-2 mandelbrot]$
```

Summary

What did we learn?

- How to make a program
- Variables
- Functions
- Control structures (for)