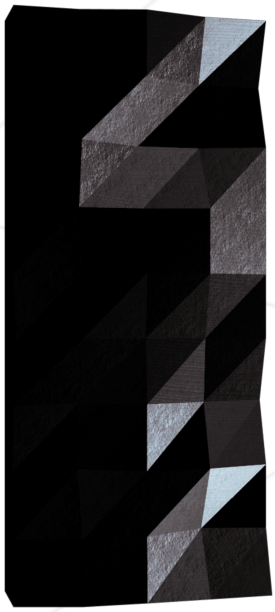

Rubik

What are you doing next Wednesday?

François Sechet fsechet@student.42.fr
42 Staff pedago@staff.42.fr

Summary: *This project will make you write a program that solves the Rubik's cubes in the least amount of moves.*

Contents

Chapter I

Foreword

During defense, you will have to be better than this:
https://www.youtube.com/watch?v=eQH7MU0gUCQ

However this is still an acceptable performance:
https://www.youtube.com/watch?v=_v85dcvw2vQ

That however is too easy:
https://www.youtube.com/watch?v=X0pFZG7j5cE

Chapter II

Introduction

In this project, we will ask you to solve Rubik's Cubes, according to the criterias of the official competitions of Rubik's Cube and in particular those from Fewest Moves Challenge (FMC), minus the usual time limit of one hour (your time limit will be counted in seconds).

During a FMC competition, all the competitors are locked in a room with a piece of paper, a pen, a cube and that's it. During 60 minutes, they individually look for the fastest solution to a given mixed cube and return it on paper. The solutions are then individually checked, and the competitor with the least moves is declared winner.

Good luck!

The current FMC world record is 20 moves, by Tomoaki Okayama.
More than 200 people have already found results below 30 moves in competition.
Also, it is proven that in a worst case scenario, for each cube there is at least one solution with a max of 20 moves. More information on: http://www.cube20.org/

You are strongly encouraged to learn how to solve Rubik's Cubes IRL for this project.

Chapter III

Objectives

Everyone knows the Rubik's Cubes, and they is a strong probability that you already bounced some brain cells on it at least once in your life.

Whether you were able to solve it or not, this project is at the level of any enthusiastic coder. It is indeed a simple algorithm project, with a bit of space representation on top, some vague group theory notions, and a bit of brain power... peanuts really.

Chapter IV

General Instructions

- This project will be corrected by humans only. You're allowed to organise and name your files as you see fit, but you must follow the following rules.
- The language is up to you. (C, C++, Perl, Python, Java, Brainfuck if you want).
- You have to handle errors carefully. In no way can your program quit in an unexpected manner (Segmentation fault, bus error, double free, etc), and particularly since it is not what's the hardest to do in this project.
- If you are working in a compilable language (C/C++ for ex) you will submit a `Makefile`. This `Makefile` must have all the usual rules. It must recompile and re-link the program only if necessary.
- You are allowed to use any library that you can justify in defense. Naturally any use of a library or external resource that does the job for you won't be accepted, you will have to justify your own algorithm in defense.
- You must be able to explain your algorithm with words and simple visual concepts.

Chapter V

Mandatory part

Your program must accept “a mix” in parameter.

“A mix” is a sequence of movements separated by one or multiple spaces to apply on a cube so to mix it not randomly but according to a pre-defined pattern.

The notation used is the international notation (F R U B L D for Front/Right/Up Back/Left/Down). To understand this notice, I redirect you to http://www.rubiksplace.com/move-notations/ which I think is a good start. The rest of the site is also quite rich in information for this project. I invite you to really watch the video of this project as well.

F B R L U D represents both a face and the move applied to this face, but the nuance is generally quite explicit. Your programs will have to accept the move modificators “’” and “2” and send back a solution using the same notation.

Here is an example of a valid sequence: R2 D’ B’ D F2 R F2 R2 U L’ F2 U’ B’ L2 R D B’ R’ B2 L2 F2 L2 R2 U2 D2

`Middle Layer Turns M, E et S as well as Cube Rotations x, y et z are not accepted.`

Your program will have to return on the standard output the movement sequences to be applied to the 3x3x3 mixed cube with the given sequence.

For example:

```
$>./rubik "F R U2 B' L' D'" | cat -e
```

Returning the opposite of the mix (or a direct variation with sequences inserted without effect to modify the length) is dangerously close to cheating...

As always you must manage the errors in a sensible manner.

The metric used to count the moves is a half-turn metric (HTM), ie a quarter turn or a half turn on the same face is equivalent to one move, but the side movements count for 2 moves. Basically your score will be the result of a “wc -w” on your solution.

```
$> ./rubik "F R U2 B' L' D'" | wc -w
```

Chapter VI

Bonus part

As long as the mandatory part rules and the general instructions are respected, you can always add all the bonuses you wish, they will be graded directly by your corrector.

For example:

- A genuine graphic representation of the cube as it progresses (ncurses, minilibx, open GL? Or whatever you want, a simple serie of numbers or letters is considered as debug, not quite bonus worthy...) also a visual representation of the faces that turn in real time (that would be something!)
- An algorithm that gets down to the limit of the fastest solutions in a delay that remains reasonable (above a few seconds, it no longer is).
- The choice between multiple algorithms, or a selection of the best solution on multiple algorithms.
- An integrated mix generator where we could specify the length and/or a number.
- A subdivision of your solution in “humanly understandable” steps.
- It works with other puzzles (4x4x4, 2x2x2, Megaminx, Square-1 ?)
- Some genuine Rubik’s Cube performance from the corrected in defense
- ...

Bonuses will be taken into account only if the mandatory part is flawless.

These bonuses cannot overwhelm the normal functions of the program (ie the input of mixes must always be a valid sequence), some eventual options can be indicated with a “-” preceding them.

Chapter VII

Submission and peer-evaluation

The correctors will be particularly cautious in case of cheating which brings forward, in official competitions, a banishment that can last years. At 42 we will simply subject the culprit to a public humiliation and he/she will have to provide sweets to the entire cluster.

Contrary to your colleagues in FMC competitions, you are allowed to check Google for help, the peer of your right AND on your left or the forum. But as per your colleagues during FMC, try first to use your brain...