Chapter I

Warnings

COREWAR ISN'T YOUR ORDINARY PROJECT

   THE SUBJECT IS INCOMPREHENSIBLE

   THE GUY WHO CREATED THIS PROJECT IS INSANE

   YOUR CHAMPION WILL NEVER BEAT RAINBOWDASH

   I REPEAT COREWAR IS NOT AN ORDINARY PROJECT

   THE SUBJECT ISN'T ENOUGH TO UNDERSTAND EVERYTHING

   COREWAR IS DEFINITELY NOT AN ORDINARY PROJECT

`Yes`, we know it all, you should always keep that in mind. Corewar is a difficult and long project that will require a determination as great as `Kwame`'s Boots. The `worse case scenario` would be to try to do the project the last week which would end in a garanteed failure.

Don't work alone. Talk to each other. Share your ideas. Unpack the subject with your peers. Debate your interpretations. Take esponsibility for your choices. Test your work. Experiment with reference binaries. Use the provided visualizer. Don't use to much drugs. Offer the staff candies, preferably schoko-bons.

Even if your program is 20 000-line long, the consequences for a norm error will exactly be same. Peer2peer correction will be `merciless`, so be `mercilessly rigorous` in your work.

Chapter II

Purely technique notions

- You already know everything in C. Corewar isn't a project that will make you discover new concepts.

- It would be clever to gather some knowledge about:

  - Parsing
  - Overall compilation (What is it, your assembler program, in your opinion ?)
  - A virtual machine

Chapter III

Ressources

You have at your disposal reference programs and some tips that'll proove precious:

- An `assembler` as well as a `virtual machine` already compiled.

- Couple of basic but functional `champions`.

- The `virtual machine` has a `-n` option that allows to have a `ncurses` view allowing you to see the game better.

The assembler, the VM and the champions are available on `sgoinfre` in a folder called `CoreWar`. On some aspect, the virtual machine and the assembler are doing a little more than what's required for the mandatory part (more options), you don't need to do as much. But i advise you to explore those options, it'll help you better understand the project.

Some important information:

- The compute part of the VM can be long. It's supposed to be like that. A lot of processes = a lot of compute time.

- You ask questions and give feedbacks on the forum, slack etc.