C Piscine

Rush-02

Contents

Chapter I

Introduction

- Your group is already registered for the defense slot — no need to worry about that. Just don't cancel it because you won't get another one.

- Keep it simple - asking questions might make things more complicated.

- Follow the submission procedures for all your exercises (you know the drill).

- Make sure your program compiles! No compile = zero points. Pretty straightforward.

- All heap-allocated memory must be properly freed when necessary. Memory leaks will not be tolerated.

- Everyone needs to show up for the defense with your assigned team.

- Make sure everyone understands the whole project, even if you split up the work. Your final grade depends on whoever explains things worst during defense, so get everyone up to speed.

- Staying in touch with your team is on you. You've got phones, email, carrier pigeons, bees, whatever... so no excuses about communication issues. Life isn't always fair, but that's just how it goes.

- If someone's completely MIA despite your best efforts, do the project anyway. We'll figure something out during defense. You'll still have access to submit even if the team leader goes missing.

- And of course, follow the Norm. Be thorough. And try to have some fun with it!

Chapter II

Foreword

In the year 2542, as the burned remains of World War III ushered in a fragile new beginning, Earth rose once more from the ashes of a nuclear wasteland. Hidden aristocrats emerged from their secret bunkers, resplendent in opulence, to reclaim a land both scarred and grim.

Across this ravaged realm, humanity had been twisted into ghastly forms: ghostly skeletons wandered like lost souls, while ugly green goblins — reminders of what once was — stalked the barren wilds.

Driven by a perverse desire for luxury and brutal entertainment, these aristocrats built vast arenas of horror. Within these bloodstained colosseums, condemned creatures were forced to engage in a savage dance for survival, their suffering displayed as a repulsive show for those who delight in cruelty.

Dear Strategist, this letter is not a request, it's an order — a call to arms on the eve of battle. Your destiny, and indeed your very life, are bound to the fate of these warriors.

The time has come to gather your forces with precision and courage, turning every command into a stroke of genius. Now, as you stand on the brink of combat, let your resolve shine like a guiding light.

Embrace the challenge and let your code strike true like a finely honed blade.

Onward — to glory!

Chapter III

Subject

	Rush 02 - Exercise 00
Program name	bot
Turn-in directory	./
Files to turn in	Template files, src/*.c, include/*.h
Allowed functions and libraries	`malloc`, `calloc`, `realloc`, `free`, `rand`, `srand`, `time`, `write`, `printf`, `<limits.h>`, `<math.h>` All function of the connection library

- You are building a bot for COREGAME using the 'core_lib.h' library.

- The display name in the visualizer should be the intra-name of your group leader. *Hint: "YOUR TEAM NAME HERE"*

- During evaluation, your bot will compete against enemy bots. Your points will depend on its performance in these matches.

- You must have a `GitHub` account to do this project.

- You will need these resources:

    - `Trailer`
    - `Introduction`
    - `COREGAME website`
    - `COREGAME wiki` (make sure you use the right version)

- To get started you have to do the following:

    - Go to `https://coregame.de` (login with your GitHub).

    - Create a team and invite your teammates.

    - Clone your repo with ssh and NOT https.

    - Type in `make devcontainer`.

    - Wait until VSCode started.

    - Type `make` in the VSCode terminal from previous step.

    - Open your browser at `http://localhost:4000`.

    - Have fun!

- Want to see if your bot is any good? Challenge other teams and see how it holds up. *Hint: Your enemy is named "gridmaster"*

Chapter IV

Bonus

- Expert mode: don't let your core take any damage at all.

- At the tournament in the end you can earn some extra points.

- If you find a bug within the game you will get 5 extra points to your score, if the code that you submit actively exploits a bug in the game, you will get -42 points. To get the extra points you need to write a documentation of the bug and send it to apregitz on discord.
  The documentation should contain the intra name of your team leader, what the bug is, the exact time when it occours and the git commit.