Operating Systems 1 – Project A 2 weeks

Thread Snake

Write a multithreaded auto-playing snake game. The program takes multiple command-line arguments: options modifying the default behaviour and at least one snake parameters. Default option values are given after equal sign:

`./tsnake [-x` x_{dim}`=20] [-y` y_{dim}`=10] [-f file=$SNAKEFILE]` c_1`:`s_1 `[`c_2`:`s_2 `...]`

The snake parameters are in form c_i:s_i where c_i is the snake's character (has to be uppercase and unique) and s_i is the snake's speed in milliseconds. For instance, the user should be able to create three snakes with different speed using the following invocation:

```
./tsnake A:500 B:1200 C:200
```

THE MAP

The main thread at the very beginning creates a 2-d array of characters with dimensions given via CLI (x_{dim}, y_{dim}) representing the game's map and fills it with randomly placed food tiles. Empty tiles are just spaces. Food tiles should be lowercase "o" characters. Tiles occupied by snake should contain the snake's character – head in the uppercase, the rest in the lowercase. The number of food tiles and the number of snakes should always be equal.

During the game the threads shall control the movement of snakes around the map so the map structure has to be properly protected by a locking scheme of your choice.

THE SNAKE THREADS

For each snake parameter a snake thread is spawned, responsible for moving a single snake around the map. At the very beginning, the thread shall put a 1-tile long snake at a random free tile. The thread always maintains a target tile – the one which the snake is moving towards. The target tile is always randomly chosen out of the set of the tiles containing food (at the moment of making choice). After reaching the target tile, a new target is chosen.

The snake thread moves the snake in a loop with delay interval defined by snake's speed parameter. In each step the snake's head moves one tile towards the target (in either the x or y axis and the tail follows. After stepping on a food tile the snake becomes longer, new food tile is generated in the snake's thread and a new target is chosen. Snakes cannot move through the map boundary. They also cannot enter tiles occupied by other snakes.

THE MAIN THREAD

The main thread after creating the snake threads reads commands from the standard input. It also reacts to some signals from the user.

When user types **show** it shall print map's state, i. e.:

```
> print
x----------------x
| o     Aaa      |
|                |
| b              |
| b              |
| bbB   o        |
x----------------x
```

When user types **spawn c_n:s_n** it spawns a new snake thread as if it was given by CLI arguments.

When user types **save** it shall save map to the file given by -f argument (in exactly the same textual format as described in show command).

When user types **exit** or sends SIGINT (C-c) via terminal the snake threads are cancelled and the program exits cleanly.

THE SAVE FILE

If the program was started without -f option the save path should be read from `$SNAKEFILE` environment variable. In case it's not present either, the program won't be able to save and user should be informed about that at start.

If the save file exists when the program starts, it should load the map state from it and verify that the CLI snake parameters correspond to the map's state – snake characters shall match exactly. The map dimension parameters are not required when loading map from file.

UPLOAD

Please upload your solution to: **/home2/samba/sobotkap/unix/**

```
scp user.tar.bz2 user@ssh.mini.pw.edu.pl:/home2/samba/sobotkap/unix/
```

Please name your stages files according to the schema: `LOGIN.tar.bz2(.gz)` Solutions not following the naming convention may not be checked.

THE STATEMENT

By decree 27/2020 of University Rector you must add the following statement to the uploads:

```
------------------------------------------------------------------------
I declare that this piece of work which is the basis for recognition of
achieving learning outcomes in the OPS1 course was completed on my own.
[First and last name] [Student record book number (Student ID number)]
------------------------------------------------------------------------
```

Please add it as comment at the beginning of each source file you upload. Replace square brackets with your data.