

[Streami] BackEnd Engineer 채용 Pre-test

Conway's Game of Life 를 visualize 하는 프로그램을 작성하십시오.

(Game of Life 에 대해서는 다음 링크를 참고하십시오: <https://bitstorm.org/gameoflife/>)

추가적인 스펙은 다음과 같습니다.

1. 전체 board 의 기본 크기는 최소 width 80, height 40 이상이어야 합니다.
 - 초기 board 에 대한 input 을 받지 않으면 random 한 board state 으로 initialize 하십시오.
 - .txt 파일로 초기 board state 을 받을 수 있도록 합니다.
 - 파일의 형식은 다음과 같습니다. (예시로 plus.txt 를 첨부합니다)
 - 첫줄 NUM_ROWS NUM_COLS
 - 두번째 줄 NUM_CELLS_INIT
 - 세번째 줄 ~ NUM_CELLS_INIT + 2 줄 cell 들의 위치 (row, col)
2. 언어에 대한 제한은 없습니다. 가장 편한 언어를 사용하여 구현하시기 바랍니다.
3. visualization 방식에 대한 제한은 없습니다. (어떤 GUI package 든 상관없고, stdout 을 사용하여 구현하셔도 무방합니다)
4. 설치 스크립트까지 포함하여 Ubuntu 16.04 에서 돌아갈 수 있는 상태로 제출되어야 합니다. (Docker/Vagrant 등 사용 가능)
 - Ubuntu 16.04 기준으로 추가적인 설정 없이 하나의 .sh 파일을 실행하면 바로 필요한 셋업이 완료되어야 합니다.
5. GitHub repository 의 URL 을 제출하십시오.
 - 부분 작업 단계마다 commit 을 남겨 개발 과정을 볼 수 있어야합니다.
 - commit message 들은 다른 분과 함께 작업을 하신다는 생각으로 진행 사항을 파악할 수 있도록 남겨주시기 바랍니다.
 - root directory 의 README.md 파일에 결과물에 대한 overview 와 실행 방법에 대해서 기술하십시오.
6. 실행 시 command line 을 통해 arguments 들을 넣을 수 있어야합니다.
 - 첫번째 argument 는 string 으로 initial board state 을 나타내는 .txt 파일을 넣을 수 있어야 합니다.
 - 두번째 argument 는 int 로 initial board state 의 .txt 파일이 놓여진 상태에서 'n'이라는 숫자가 입력됐을 시 'n' generation 이후 board state 를 초기 board state 를 나타내는 txt 파일과 같은 형식으로 dump 합니다. (초기 상태는 0 generation 입니다)
 - 즉, 총 세 가지 방법을 통해 실행할 수 있어야 합니다 (예시: Python 의 경우)
 - python game_of_life.py -- random board state 로 시작하고 여러 generation 에 걸쳐 board 가 바뀌는 것을 visualize 한다
 - python game_of_life.py plus.txt -- plus.txt 에 주어진 state 로 시작해서 generation 에 걸쳐 board 가 바뀌는 것을 visualize 한다
 - python game_of_life.py plus.txt 10 -- plus.txt 에 주어진 state 로 시작해서 10 generation 후의 board state 을 txt 파일로 dump 한다.

궁금하신 부분에 대해서는 질문을 주시면 최대한 빠르게 답변드리도록 하겠습니다.