

Zgadywanie liczb

Gra logiczna

14.06.2022

Marian Buzak

Spis treści

1. Założenia programu.....	3
2. Pliki programu.....	3
3. Rozgrywka.....	3
3.1. Nowa gra.....	4
3.2. Poziom gry.....	4
3.3. Wyniki.....	4
3.4. Wyjście.....	4
4. Grywalność.....	4

1. Założenia programu

W notesie zajęć w Teamsach były opisane poziomy do tej gry od 1 do 8. Pisząc swój program wzorowałem się właśnie na tych punktach, ale nie w pełnym wymiarze. Część programu zaimplementowałem wykorzystując własne pomysły, które opiszę w dalszej części dokumentacji. Oprócz tego program jest możliwy do uruchamiania przy użyciu systemu Windows oraz Linux.

Program zawiera

1. Zrealizowany poziom 1 oraz 2
2. Poziom 3. Program nie pyta o ponowne zaczęcie rozgrywki, gdyż zawiera 7 rund, które są w kolejności uruchamiane. Dla każdej rundy jest losowana nowa liczba. Aby zacząć jeszcze raz należy w menu wybrać opcję „Nowa gra”
3. Poziom 4. Wyniki są zapisywane w klasie Scoreboard, która w momencie zakończenia gry przez użytkownika jest aktualizowana. Aby wyświetlić wyniki, należy w menu gry wybrać opcję „Wyniki”. Posortowana lista wyników zawiera: imię, ilość ukończonych poziomów oraz rangę. Dostępne są rangi w kolejności: król, królowa, wieża, gонец, skoczek, pionek
4. Poziom 5. Wyniki są zapisywane do pliku przy wyjściu z programu oraz są wczytywane przy uruchomieniu
5. Poziom 6. Grę testowałem samodzielnie, głównie na systemie windows
6. Gra zawiera dodatkowo menu gry, limity czasowe oraz limity ilości prób

2. Pliki programu

Aby program działał poprawnie należy mieć pobrane pliki:

1. main.rb – główny plik programu, zawiera również klasę Menu
2. game.rb – plik zawierający klasę Game, w której jest zawarta logika gry
3. score.rb – plik zawierający klasy Score oraz Scoreboard oraz pojedynczy test działania. Klasa Scoreboard zawiera listę klas Score
4. scoreboard.txt – plik tekstowy służący do zapisu/odczytu tablicy wyników

3. Rozgrywka

Użytkownik uruchamia program z poziomu terminala wpisując polecenie „ruby main.rb”. W tym momencie użytkownikowi wyświetla się menu gry, w którym ma do wyboru jedną z opcji: Nowa gra, zmiana poziomu trudności, wyświetlenie tablicy z wynikami oraz wyjście z programu. Również podczas uruchomienia programu jest wczytywana historia zapisanych wyników z pliku scoreboard.txt.

3.1. Nowa gra

Na pojedynczą rozgrywkę składa się 7 rund. Taka liczba rund jest ustawiona w celu, aby użytkownik nie zgadł liczby np. za pierwszym razem i ten wynik nie został zapisany na najwyższej pozycji.

W każdej rundzie losowana jest liczba w zakresie $<1, 128)$, na której znalezienie użytkownik ma ograniczoną liczbę prób i czasu. Od użytkownika oczekiwanym wejściem jest liczba całkowita lub napis „koniec”, który kończy grę. W przypadku, gdy użytkownik poda niewłaściwe wejście program wypisze komunikat błędu „liczba poza zakresem” lub „niepoprawna wartość”. Niewłaściwe wejście nie zwiększa licznika ilości prób.

W momencie, gdy ilość prób zostanie wyczerpana lub czas się skończy, zostaje wyświetlany napis „przegrana”. Docelowo użytkownik powinien ukończyć wszystkie 7 rund, po których zostanie wyświetlony napis „zwycięstwo”. Niezależnie czy użytkownik wygrał lub przegrał, ma możliwość zapisania swojego imienia do tablicy z wynikami – jedynym warunkiem jest osiągnięcie przynajmniej 2-giej rundy.

Jeśli przy wpisywaniu wyników, pole z imieniem zostanie pominięte, zostanie przypisane imię „anonim”.

3.2. Poziom gry

Opcja zmiana poziomu trudności pozwala użytkownikowi wybrać jeden z poziomów: łatwy, normalny lub trudny. Każdy z poziomów trudności zawiera dwa parametry, dzięki którym są obliczane: maksymalna ilość prób zgadnięcia liczby na rundę oraz limit czasowy.

3.3. Wyniki

Wyświetlanie wyników pokazuje najlepsze 10 wyników posortowanych po poziomie trudności oraz ukończonych rundach. Wyniki są aktualizowane w momencie, gdy użytkownik kończy grę. Pole rangi jest obliczane przy użyciu poziomu trudności oraz liczby ukończonych rund. Czas nie jest tutaj użyty, gdyż czas jest zmniejszany przy każdej następnej rundzie. Symbolami rang są figury szachowe(hobby).

3.4. Wyjście

Zakończenie programu równocześnie czyści konsolę oraz zapisuje zaktualizowane wyniki do pliku scoreboard.txt.

4. Grywalność

Problem grywalności polega na tym, aby znaleźć optymalne ograniczenia ilości prób zgadnięcia liczby oraz limitu czasowego, żeby gra była wymagająca do ukończenia dla użytkownika. Aby ten problem rozwiązać, należało wymyślić wzór na wyliczanie liczb ograniczających oraz przetestować go na kilkunastu przypadkach.

W programie jest zaimplementowany wzór, którego wynik przedstawię na poniższym przykładzie.

Założmy, że program będzie generował liczby w zakresie $<1, 1024)$ na przestrzeni 4 rund.

Użytkownik wtedy jest w stanie znaleźć wylosowaną liczbę przy 10 podejściach, ponieważ $\log_2 1024 = 10$.

Poziom trudności	Pierwsza runda(czas)	Ostatnia runda(czas)	Ilość prób
Łatwy	$10 \cdot 16 = 160s$	$10 \cdot 8 = 80s$	$10 \cdot 4 = 40$
Normalny	$10 \cdot 8 = 80s$	$10 \cdot 4 = 40s$	$10 \cdot 2 = 20$
Trudny	$10 \cdot 4 = 40s$	$10 \cdot 2 = 20s$	$10 \cdot 1 = 10$

Pozostaje jeszcze do obliczenia czas dla np. 2-giej lub 3-ciej rundy. W tym celu użyłem następujący wzór $czas - ((runda / (rundy - 1)) * diff)$, gdzie

runda oznacza obecną rundę(0, 1, 2), które liczone są od zera,

rundy oznacza ilość rund(4),

diff jest różnicą czasu między pierwszą i ostatnią rundą,

czas oznacza czas przeznaczony w pierwszej rundzie

Czyli np. 2-ga runda dla poziomu normalnego będzie miała ustawiony czas $80s - ((1 / (4-1)) * 80s - 40s) = 80s - (1/3 * 40s) = 80s - 16,6s \rightarrow 80s - 17s = 63s$