Politechnika Śląska w Gliwicach

Wydział Informatyki, Elektroniki I Informatyki

**Podstawy Programowania**

**Komputerów**

**Gra w wisielca**

Autor Wojciech Mizia

Prowadzący Prof. dr hab. Inż. Wojciech Sułek

Rok Akademicki 2020/2021

Kierunek Teleinformatyka

Rodzaj Studiów SSI

Semestr 1

Termin laboratorium Wtorek,czwartek 08:00-10:15

Grupa 4

Sekcja 10

Termin oddania sprawozdania 2020-12-17

Data oddania sprawozdania 2021-01-09

* **Treść zadania**

Program, będący implementacją gry towarzyskiej w wisielca, powinien umożliwiać wybór kategorii losowanego hasła oraz przechowywanie statystyk o wygranych i przegranych grach poszczególnych graczy. Kategorie wraz z hasłami powinny być przechowywane w pliku tekstowym. Podczas odgadywania hasła na ekranie (w oknie konsoli) powinna być rysowana szubienica wraz z wisielcem. Program uruchamiany jest z linii poleceń z wykorzystaniem następujących przełączników:

-i plik wejściowy z kategoriami i hasłami

-w plik wejściowy/wyjściowy zawierający statystyki wygranych/przegranych gier

poszczególnych graczy

Uruchomienie programu bez parametrów powoduje wypisanie krótkiej informacji o tym, jak użyć programu.

* **Analiza zadania**

1. Zagadnienie przedstawia problem podmiany odpowiednego znaku, podczas gdy użytkownik trafi w literę znajdującą się w zgadywanym haśle lub naliczenie blędnej próby, gdy nie uda mu się trafić.

2. Odpowiednie wylosowanie hasła w zależności od kategorii (wylosowanej lub wybranej przez użytkownika).

3. Zapisywanie,odczytywanie oraz odpowiednia zmiana statystyk danych graczy w zależności od wygranej lub przegranej.

**2.1 Algorytmy**

Program posiada menu pozwalające wybrać jedną z pięciu akcji

1.Rozpocznij rozgrywkę

Podczas rozgrywki program dla danej kategorii losuje odpowiednie słowo po czym ukazuje nam gre oraz czeka na podanie znaku, po podaniu dodaje znak do *wszystkie\_strzaly*, jeśli ten znak się tam nie znajduje. Kolejny krok to sprawdzenie ile błędnych znaków podał gracz, gdy gracz podał zły znak 6 razy rozgrywka się kończy oraz dodaje graczowi przegraną w statystykach. Następnym krokiem programu jest sprawdzenie czy gracz wygrał, w tym momencie funkcja “zgadywanie” sprawdza czy w naszych wszystkich strzałach znajdują się pojedyńcze znaki hasła oraz rysuje nasz postęp zgadywania (znak słowa który został dobrze trafinony jest wyświetlany w sytuacji odwrotnej za ten znak wstawiane jest “\_ ”, jeśli wszystkie znaki słowa zostaną odnalezione funkcja zgadywanie zwraca “true”. Następnie program sprawdza czy wygraliśmy, jeśli tak to wychodzi z pętli i dodaje wygraną danemu graczowi.

2.Wybor kategori

Pozwala wybrac jedną z danych kategorii znajdujących się w pliku

*kategorie\_hasla.txt*.

3.Statystyki graczy

Pokazuje nam zawartosc pliku *statystyki.txt*.

4.Wybierz ilosc graczy

Pozwala nam zmienić ilość graczy od 1 do 8 (domyślnie 4).

5.Wyjdz z gry

Konczy program.

* **Specyfikacja zewnętrzna**

Program jest uruchamiany z linii poleceń. Należy przekazać programowi nazwy plików: wejściowego i wejściowo-wyjściowego po odpowiednich przełącznikach(odpowiednio -i oraz -w) np.

Program -i *kategoria\_hasla.txt* -w *statystyki.txt*

Pliki wejściowe i wejściowo-wyjściowe są plikami tekstowymi z rozszerzeniem .txt. Nieprawidłowe podanie przełączników wyświetli komunikat:

*Zly format!*

oraz wyświetli jak poprawnie wprowadzić dane.

Poprawne uruchomienie programu wyświetli proste menu z pięcioma możliwymi wyborami. Podczas działania programu wprowadzenie nieodpowiednych znaków wyświetli odpowiednio komunikaty:

*To nie litera!*

*Podana opcja nie istnieje*

*Wprowadz poprawna liczbe!*

Jeśli wystąpi błąd z plikami, z których program korzysta, nastąpi wyświetlenie komunikatu

*Blad otwarcia pliku!*

,lub jeśli plik jest pusty

*Pusty plik!*

*Program nie zawiera polskich znaków.*

Istnieje możliwość dodania kategorii oraz haseł do biężacych kategorii w pliku *kategoria\_hasla.txt*. Należy jednak pamiętać o tym że: każda linia powinna zawierać jedno słowo, hasła do danej kategorii powinny znajdować się pod kategorią do której należą, kategorię poprzedza znak “-“, oraz plik musi być zakończony słowem *!KONIEC!.*

* **Specyfikacja wewnętrzna**

Funkcje programu:

*Int pobierz\_liczbe(int poczatek, int koniec)*

Pobiera liczbe z przedziału *poczatek*, *koniec* jako pojedyńczy znak*.* Sprawdza czy podany znak jest poprawny, jeśli nie to prosi o ponowne podanie. Po podaniu poprawnego znaku przypisuje zmienia go na inta. Pod koniec drukuje którą liczbe wybraliśmy: *Wybrales (liczba) liczbe!*

*void wiadomosc(string wiadomosc, bool czy\_przerwa = false)*

Wyświetla wiadomość pomiędzy dwoma ściankami( | wiadomosc | ) zachowuje zawsze odpowiednia długość dzięki pętli, która zaczyna się od długości wiadomości jaką przekazaliśmy funkcji do 40 znaków. Od bool`a zależna jest przerwa nad naszą wiadomością, żeby odzielić ją od reszty na ekranie.

*string wybor\_kategori(string plik, bool czy\_losowo=false)*

Funkcja pierwsze pobiera dane z pliku txt., następnie zależnie od bool`a *czy\_losowo* losuje liczbe z przedziału od 1 do ilości kategorii lub pozwala ją wybrać użytkownikowi. Kolejnym krokiem tej funkcji jest znalezienie odpowiedniej kategori w zależności od wylosowanej lub wybranej liczby i zwróceniu jej.

*int ile\_prob\_do\_konca(string slowo, string strzal)*

Sprawdza ile niepoprawnych strzałów wprowadził użytkownik. Wyszukuje w haśle kolejne znaki stringa *wszystkie\_strzały*, jeśli nie znajdzie danego znaku to *bledne\_proby* inkrementują się.

*void wisielec(int bledne\_proby)*

Rysuje kolejne części wisielca odnosząc się do funkcji wiadomosc.

*bool zgadywanie(string slowo, string strzal)*

Funkcja rysuje hasło na ekranie w postaci podłogi ( \_ ), jeśli literka nie została odgadnięta lub rysują ją w przeciwnej sytuacji. Sprawdza ile znaków hasła znajduję się w *wszystkie\_strzały*, jeśli wszystkie to zwraca prawde.

*void rysuj\_litery(string dane\_litery, char poczatek, char koniec)*

Rysuje litery jakich jeszcze nie wykorzystał użytkownik lub podłogę ( \_ ), jeśli użytkownik już wykorzystał daną literę. Początek oraz koniec przekazuję jej funkcja *dostępne\_litery*.

*void dostepne\_litery(string wszystkie strzaly)*

Przekazuje funkcji *rysuj\_litery* od której litery należy zacząć, a na jakiej skończyć rysowanie danej linii liter.

*string losuj\_slowo(string plik, string kategoria)*

Losuje słowo w zależności kategorii. Funkcja pierwsze sprawdza ile haseł posiada dana kategoria, a następnie losuje wartość od 1 do ilości haseł i zależnie od tej wartości szuka odpowiedniego hasła.

*void statystyki(string plik, int nr\_gracza, bool czy\_wygral=true, bool rysuj = false)*

Funkcja operuje na pliku *statystyki.txt*. Pierwsze wczytuje statystyki danych graczy, a następnie w trakcie działania programu inkrementuje wygrane lub przegrane, zależnie od tego czy użytkownikowi udało się odgadnąć hasło.

* **Testowanie**

Użytkownik zawsze zostaje powiadomiony gdy wprowadzi niepoprawne dane stosowną wiadomością, więc jest prosty w obsłudze. Możliwość wprowadzenia błędnych danych istnieje tylko przy uruchamianiu programu, lecz również użytkownik zostaja powiadomiony o istniejącym błędzie. Program został przetestowany na pustych plikach oraz gdy te pliki nie istniały, takie sytuacje powodują zgłoszenie błędu.

* **Wnioski**

Program do gry w wisielca jest bardzo prosty, lecz wymagającym okazał się sam algorytm rozgrywki, który odpowiada za całą rozgrywkę i komunikację z użytkownikiem. Plik wejściowy programu (plik txt), można rozbudowywać o nowe kategorie oraz hasła, więc program daje możliwości kompletnej zmiany kategorii oraz haseł.