Politechnika Śląska

Wydział Matematyk Stosowanej

Kierunek Informatyka

Gliwice, 03.02.2022

Programowanie I

**projekt zaliczeniowy**

**"*Zestaw mini gier*"**

**Julia Jędrzejczyk gr. lab 3/5**

1. **Opis projektu.**

Projekt składa się z 3 części;

1. Gra „Kółko i krzyżyk”
2. Quiz językowy
3. Gra w kości

Całość jest opatrzona w menu, w którym dodatkowo widnieje opcja „wyjścia z programu”.

1. **Wymagania**

1. W grę „Kółko i krzyżyk” będą mogły grać 2 osoby, każda z nich będzie wybierać odpowiednie miejsce na planszy, po czym program wpisze symbol przypisany do danego gracza do wyznaczonego pola. Dodatkowo wyświetlona będzie dokładna instrukcja, program poda zwycięzcę i umożliwi zaczniecie gry od nowa.

1) Wyświetlenie instrukcji

2) Wygenerowanie planszy

3) Umożliwienie wpisania odpowiedniego symbolu

4) Umożliwienie wybrania odpowiedniego pola na planszy

5) Sprawdzenie wszystkich możliwości wygranych (w tym remisu)

2. Na początku programu zostanie ukazana instrukcja gry. Użytkownik zostanie poproszony o wpisanie podanego słowa, ale po angielsku. Zostanie poinformowany czy jego tłumaczenie jest dobre czy też nie. Jeżeli popełni błąd, na ekranie wyświetli się poprawna wersja. Jego punkty zostaną zsumowane pod koniec gry, a wynik zostanie wyświetlony na ekranie.

1) Wyświetlenie instrukcji

2) Poproszenie o wpisanie tłumaczenia odpowiedniego słowa

3) Sprawdzenie odpowiedzi, ewentualne wyświetlenie poprawnej odpowiedzi i podanie informacji zwrotnej

4) Zliczanie punktów

5) Wyświetlenie informacji o zdobytej liczbie punktów

3. Wyświetlenie instrukcji i zasad gry. Gra będzie polegać na rzucie 5 kośćmi (losowanie wartości od 1 do 6), a następnie odpowiednie przypisanie punktów zgodnie z zasadami gry. Punkty będą sumowane, po zakończeniu gry, wyświetlony zostanie zwycięzca.

1) Wyświetlenie instrukcji

2) Wprowadzenie liczby rund przez użytkownika

3) Losowanie wartości oraz ich wyświetlenie dla 1. użytkownika

4) Informacja zwrotna na temat zdobytych punktów

5) Losowanie wartości oraz ich wyświetlenie dla 2. użytkownika

6) Podliczenie punktów i ogłoszenie zwycięzcy

**3. Przebieg realizacji**

Stworzone zostały 3 mini gry. Każda z nich zawiera: funkcje, część główną oraz pliki tekstowe.

Dodatkowo, po każdej wyświetlonej instrukcji została dodana funkcja *system(‘’pause’’),* która zapobiega przypadkowemu kliknięciu, aby rozpocząć program. Zapotrzebowanie na powyższą akcję, wyniknęło z testów przeprowadzonych na domownikach oraz z ogólnych wniosków, z tego w jakim sposób funkcjonujemy – szybko i często robimy, a następnie myślimy. W związku z tym w programie pojawiło się właśnie to zabezpieczenie.

Dodatkowo przy podaniu złej odpowiedzi przez użytkownika i wyświetlonej prawidłowej odpowiedzi, zdecydowałam się na taki sam krok *system(‘’pause’’)*, aby użytkownik miał czas na przeanalizowanie swojego błędu oraz naukę nowych zagadnień. Jeżeli błąd nie został popełniony, użytkownik dostaje informację zwrotną w postaci gratulacji, a program za pomocą funkcji *Sleep* zatrzymuje ową pochwałę przez pewien czas.

Pliki zawarte w programie:

2x plik z wynikami (Quiz językowy, Gra w kości)

3x plik ze słowami (Quiz językowy)

3x plik z tłumaczeniem słów (Quiz językowy)

Zawarte biblioteki:

<stdio.h> aby działała funkcja *exit*

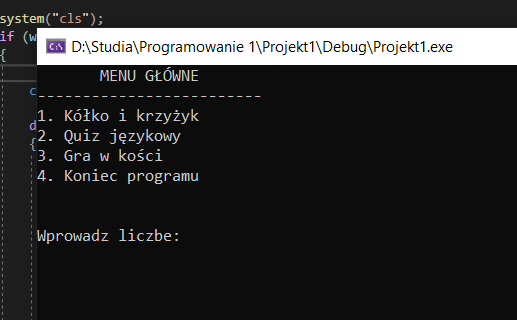
<windows.h> aby działa funkcja *cls* oraz funkcja *Sleep*

<random> aby działała funkcja losująca

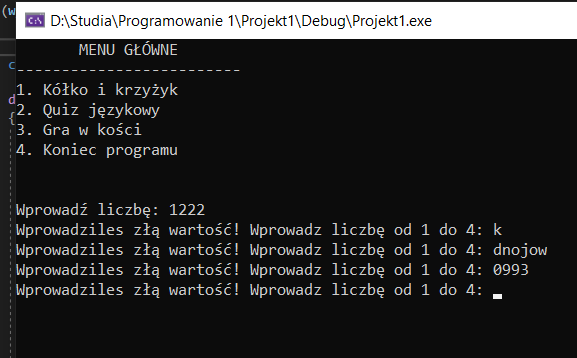
<ctime> aby działała funkcja losująca

<string> aby można było wykonywać działania na „stringach”

<fstream> aby można było wykonywać działania na plikach

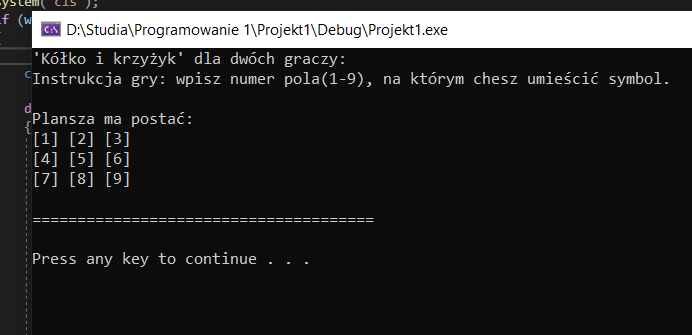
**4. Instrukcja użytkownika**

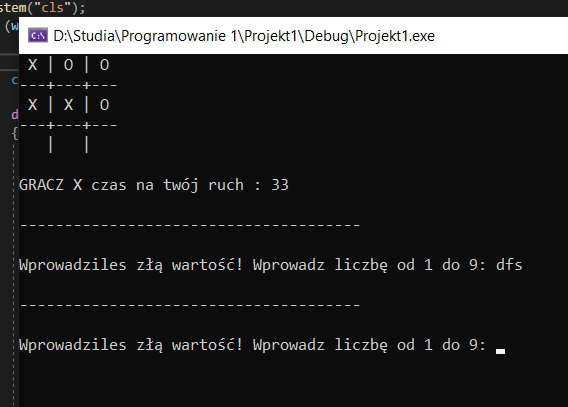
Program składa się z 4 części i menu. W menu mamy możliwość wyboru interesującej nas gry: kółko i krzyżyk, quiz językowy, gra w kości. Ostatnia pozycja menu umożliwia nam zakończenie programu.

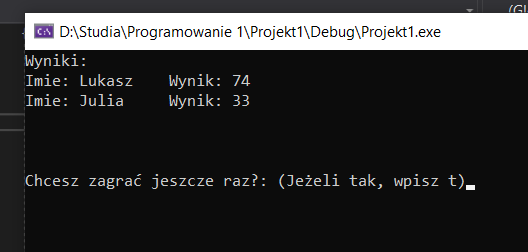
Aby wybrać interesujący nas aspekt, należy wpisać odpowiednią liczbę z klawiatury. Program jest odpowiednio zabezpieczony, aby po wpisaniu nieodpowiedniej wartości, zwrócić użytkownikowi uwagę i udostępnić mu możliwość wpisania odpowiedniej liczby raz jeszcze. Czynność ta jest wykonywana do skutku (tj. do wpisania odpowiedniego numeru). Sytuacja ta jest powtarzana na każdym etapie programu, kiedy tylko użytkownik proszony jest o wprowadzenie liczby.

Jeżeli użytkownik poprawnie wpisze numer, program automatycznie przekierowuje go do wybranej gry (lub też kończy program).

Uruchamiając grę, automatycznie zostaje wyświetlona instrukcja. Aby przejść do kolejnego etapu gry, użytkownik proszony jest o naciśniecie jakiegokolwiek przycisku w celu potwierdzenia chęci rozpoczęcia gry.





**Gra przebiega według instrukcji i poczynań użytkownika. Po jej zakończeniu na ekranie pojawia się pytanie czy użytkownik chce zagrać raz jeszcze czy też nie. Jeżeli wybierzmy opcję „nie”, następuje automatyczny powrót do menu, gdzie ponownie możemy zdecydować się na wzięciu udziału w którejś z rozgrywek lub opuszczenie programu.

W programie pojawia się wiele funkcji, które umieszczone są wraz z definicją przed funkcją *main()*. Każda z nich, w miarę możliwości nosi nazwę, która symbolizuje wykonywaną przez nią akcję/czynność.

Opis funkcji zawartych w programie:

|  |  |  |
| --- | --- | --- |
| **Gra** | **Nazwa funkcji** | **Użyteczność** |
| Kółko i krzyżyk | plansza | Tworzenie planszy |
| Kółko i krzyżyk | ruch | Funkcja odpowiadająca za kontrolowanie ruchu użytkownika i wpisywanie wartości do planszy |
| Kółko i krzyżyk | ktoWygrywa | jeśli gracz będzie miał 3 symbole pod rząd w kolumnie, wierszu lub przekątnej funkcja zwraca wartość true |
| Kółko i krzyżyk | remis | Jeśli na planszy nie ma wolnej pozycji, a funkcja ktoWygrywa nie wykryła wygranego, funkcja remis zwraca true |
| Quiz językowy | zapiszDoPliku | Funkcja zapisuje wyniki do pliku |
| Gra w kości | Struktura Gra | int tab1[5]{};  int tab2[5]{};  int tab3[5]{};  int suma = 0;  string imie1;  void losuj(int liczba);  void zliczPunkty(int liczba);  void zapiszDoPliku(string nazwaPliku, string imie1);  void odczytajZPliku(string nazwaPliku);  void wyczyscPlik(string nazwaPliku); |
| Gra w kości | losuj | losowanie liczb i zapisywanie ich do odpowiedniej tablicy |
| Gra w kości | zliczPunkty | zlicza sumę wyrzuconych oczek, jeżeli gracz wyrzucił dwie liczby obok siebie to podaną sumę z danej rundy mnoży x2 |
| Gra w kości | zapiszDoPliku | Funkcja zapisuje wyniki do pliku |
| Gra w kości | odczytajZPliku | Funkcja odczytuje wyniki do pliku |
| Gra w kości | wyczyscPlik | Funkcja „przygotowuje” plik do następnej gry, czyści jego zawartość |

**5. Podsumowanie i wnioski.**

Elementy założone, zostały zrealizowane. Systematycznie, wraz z pogłębianą wiedzą, program był aktualizowany o nowe funkcje i techniki rozwiązań.

Nie obyło się jednak bez problemów:

\*poprzez problem rozumiem napotkaną przeszkodę, którą udało mi się pokonać, jednak dopiero po wielu godzinach pracy, bezowocnych próbach i przeczytanych artykułach.

1. Gra kółko i krzyżyk:

Problem z zapętleniem programu po wpisaniu przez użytkownika wartości innej niż INT, problem z zapętleniem programu, problem z odpowiednim wyświetlaniem się planszy. Wyjściowa wersja zakładała, iż plansza będzie tablicą dwuwymiarową, jednak po rozpisaniu elementów na kartce, zdecydowałam się na tablicę jednowymiarową.

1. Quiz językowy:

Spory problem sprawiło mi wypisywanie odpowiednich słów z plików to tablic. Bo rozwiązaniu problemu, napotkałam kolejny: wskaźniki i tablice wskaźnikowe. Problem z ograniczeniem możliwości wpisywania słów oraz ograniczenie ilości podpowiedzi w zależności od poziomu trudności gry. Do teraz borykam się z problem z podziału kodu na funkcje.

1. Gra w kości

Problem z czytelnym wyświetlaniem informacji na ekranie, problem   
z odpowiednim odczytem z pliku, problem z wyczyszczeniem zawartości pliku (ostatecznie zdecydowałam się na stworzenie nowej funkcji).

Wnioski i założenia:

* Dalsze i regularne programowanie.
* Skupienie się na optymalizacji kodu.
* Pisanie funkcji i odpowiednie ich wywoływanie w funkcji głównej.
* Dbanie o przejrzystość kodu.
* Rozwój napisanego programu – kółko i krzyżyk (użytkownik vs komputer); quiz językowy (optymalizacja kodu, stworzenie funkcji, projekt graficzny); gra w kości (rozwinięcie gry o nowe zasady).