

Politechnika Śląska
Wydział Matematyk Stosowanej
Kierunek Informatyka

Gliwice, 10.02.2022

Programowanie I
projekt zaliczeniowy

"MY WATCHLIST"

Emilia Pawela gr. lab. 2/4

1. Opis projektu.

Tematyka projektu dotyczy stworzenia programu który pozwala na zapisywanie informacji oraz ich wyświetleniu po obejrzeniu filmu, serialu lub anime. Istnieje również podglądu statystyk z poszczególnych kategorii lub całej kolekcji. Dodatkowo można zrobić własny quiz sprawdzający wiedzę.

2. Wymagania

- Możliwość wyboru jednej z trzech kategorii (film, serial, anime).
- Możliwość wprowadzenia danych do kolekcji, do wyboru 3 kategorie.
- Możliwość wyświetlenia listy poszczególnych kategorii, lub całej kolekcji.
- Możliwość wyczyszczenia całej kolekcji lub jednej z trzech kategorii.
- Podgląd na statystyki takie jak np. czas oglądania, ilość odcinków.
- Interaktywne menu.

3. Przebieg realizacji

Projekt składa się z 25 plików, w tym 10 z rozszerzeniem .cpp.

Gdy program zostaje uruchomiony, ukazuje się menu, gdzie można podjąć jeden z 6 wyborów. Menu główne jest zdefiniowane w funkcji `menu_glowne()`. Następnie po wyborze jednej z dostępnych opcji (wybranie opcji z poza menu spowoduje komunikat o błędzie i prośbę o ponowny wybór.) użytkownik zostanie przekierowany do odpowiedniej funkcji. Zrealizowane przy użyciu warunku wielokrotnego wyboru.

Opcje:

1. Wyświetl listę (plik `wyświetl.cpp`)

Odpowiada za nią funkcja `wyświetl_liste()`. Dzięki niej pokazuje się kolejne menu wyboru, które po wybraniu jednej z opcji, umożliwia wczytanie z pliku kolekcji z poszczególnych kategorii lub całej. Jest również możliwość powrotu do menu głównego oraz zakończenia programu. Po skończeniu wyświetlania poszczególnej kolekcji, użytkownik zostaje zapytany czy chce wykonać inne działanie, czy zakończyć program(zrealizowane przy użyciu pętli do while).

Wykorzystywany jest odczyt z plików oraz warunek wielokrotnego wyboru switch ... case.

2. Dodaj element do listy(dodaj element.cpp)

Odpowiada za nią funkcja *dodaj_element()*. Po wyborze tej opcji, pokazuje się kolejne menu, które prosi użytkownika o wybranie opcji, a następnie o podanie informacji na temat jednej z wybranych kategorii, które następnie zostają zapisane do pliku i mogą być wyświetlane po ponownym wyborze opcji, wyświetl listę. Jest również możliwość powrotu do menu głównego oraz zakończenia programu. Po skończeniu dodawania elementu do poszczególnej kolekcji, użytkownik zostaje zapytany czy chce wykonać inne działanie, czy zakończyć program(zrealizowane przy użyciu pętli do while). Wykorzystywany jest zapis do plików oraz warunek wielokrotnego wyboru switch ... case.

3. Statystyki(plik statystyki.cpp)

Odpowiada z nią funkcja *statystyka()*. Po wybraniu, ukazuje się kolejne menu. Jest możliwość wyboru wyświetlenia statystyk z poszczególnych kategorii, które są na bieżąco w tej funkcji podliczane, lecz wprowadzone zostają do przygotowanych plików tekstowych w funkcji *dodaj_element()*. Jest również możliwość powrotu do menu głównego oraz zakończenia programu. Po wyświetleniu statystyk, użytkownik zostaje zapytany czy chce wykonać inne działanie, czy zakończyć program. Wykorzystywany jest odczyt z plików i zmiana typu zmiennej, warunek wielokrotnego wyboru switch ... case.

4. Sprawdź swoją wiedzę (plik Wybór w quiz.cpp)

Odpowiadają za nią 3 funkcje. Pierwsza pozwala dokonać wyboru(funkcja *wybor_w_quiz*) czy użytkownik chce dodać pytanie do quizu(funkcja *stworz_pytania()*), a druga w czy chce w niego zagrać(funkcja *zgraj()*). Użytkownik ma możliwość wyboru ile pytań chce dodać, lub w ile chce zagrać. Użytkownik dodając pytania jest proszony o wprowadzenie konkretnych informacji. Po uzupełnieniu informacji dla jednego, pojawia się kolejne do uzupełnienia, o ile użytkownik wybrał większą ilość. W wypadku wyboru opcji gry, po udzieleniu odpowiedzi na pytanie

użytkownik dostaje informacje zwrotną czy dobrze odpowiedział, jeśli nie to dostaje również informacje, która odpowiedź była poprawa. Naliczane są punkty podczas rozwiązywania quizu, które są wyświetlane po ostatnim pytaniu. Jest również możliwość powrotu do menu głównego oraz zakończenia programu. Po zakończeniu działania, użytkownik zostaje zapytany czy chce wykonać inne działanie, czy zakończyć program. Wykorzystywany jest odczyt z pliku, zapis do plików, dynamiczna alokacja pamięci, warunek wielokrotnego wyboru switch ... case..

5. Wyczyść kolekcje oraz quizy. (plik Wyczyść liste.cpp)

Odpowiada za nią funkcja wyczyść(). Po wybraniu, ukazuje się kolejne menu. Jest możliwość czyszczenia całej kolekcji lub poszczególnych kategorii. Po wyborze, dane z pliku zostają usunięte. Jest również możliwość powrotu do menu głównego oraz zakończenia programu. Po zakończeniu działania, użytkownik zostaje zapytany czy chce wykonać inne działanie, czy zakończyć program. Wykorzystywany jest odczyt z pliku i zmiana zawartości pliku, warunek wielokrotnego wyboru switch ... case..

6. Zakończ program

Pozwala zakończyć program.

Program jest podzielny na kilka plików, gdzie każda funkcja posiada swój własny, więc jest wykorzystywany również plik nagłówkowy.

Biblioteki:

-<iomanip> - wykorzystywane do ustawienia precyzji liczb zmiennoprzecinkowych.

-<fstream> - wykorzystywane do pracy na plikach.

-<strings> - wykorzystywane do korzystania z obiektu typu string

-<Windows.h> - wykorzystywane do kolorowego menu głównego.

-„Naglowek.h”- praca na kilku plikach.

4. Instrukcja użytkownika

Instrukcja polega na wybieraniu opcji z menu.

C:\Users\amper\source\repos\projekt programowania

```
OTO WATCHLISTA
MENU:
1. Wyświetl watch liste
2. Dodaj elementy do listy
3. Statystyki
4. Sprawdź swoją wiedzę
5. Wyczyść kolekcje oraz quizy
6. Zakończ program
Wybierz opcje:
```

C:\Users\amper\source\repos\projekt programowania

```
WYŚWIETLANIE WATCHLISTY
KATEGORIE:
1. Film
2. Serial
3. Anime
4. Całą kolekcje
5. Powrót do menu głównego
6. Zakończ program
Wybierz opcje:
```

C:\Users\Pumpe\source\repos\Projekt programowanie\Debug\Projekt programowanie.exe

*****LISTA FILMÓW*****

Tytuł: Incepcja
Studio: Warner Bros
Reżyser: Christopher Nolan
Średni czas trwania filmu w minutach: 148
Data premiery: 8 lipca 2010
Postacie: Cobb, Arthur, Ariadne, Eames, Saito
Gatunek: Surrealistyczny, Thriller, Sci-Fi
Scenarzyści: Christopher Nolan
Aktorzy: Leonardo DiCaprio, Joseph Gordon-Levitt, Elliot Page, Tom Hardy, Ken Watanabe

Tytuł: MATRIX
Studio: Warner Bros
Reżyser: Lilly Wachowski, Lana Wachowski
Średni czas trwania filmu: 136
Data premiery: 24 marca 1999
Postacie: Neo, Trinity, Morfeusz, Agent Smith, Wyrocznia, Cypher
Gatunek: Akcja, Sci-Fi
Scenarzyści: Lilly Wachowski, Lana Wachowski
Aktorzy: Keanu Reeves, Carrie-Anne Moss, Laurence Fishburne, Hugo Weaving, Gloria Foster, Joe Pantoliano

Tytuł: TRANSFORMERS
Studio: DreamWorks SKG, Paramount Pictures, Amblin Entertainment
Reżyser: Michael Bay
Średni czas trwania filmu: 144
Data premiery: 12 czerwca 2007
Postacie: Sam Witwicky, Mikaela Banes, Kapitan William Lennox, Sierżant Epps
Gatunek: Akcja, Sci-Fi
Scenarzyści: Alex Kurtzman, Roberto Orci
Aktorzy: Shia LaBeouf, Megan Fox, Josh Duhamel, Tyrese Gibson

Tytuł: THOR: RAGNAROK
Studio: Marvel
Reżyser: Taika Waititi
Średni czas trwania filmu: 136
Data premiery: 10 października 2017
Postacie: Loki, Thor, Korg, Hela, Arcymistrz
Gatunek: Fantasy, Przygodowy
Scenarzyści: Craig Kyle, Christopher L. Yost
Aktorzy: Tom Hiddleston, Chris Hemsworth, Taika Waititi, Cate Blanchett, Jeff Goldblum

C:\Users\Pumpe\source\repos\Projekt programow

DODAWANIE ELEMENTU DO WATCHLISTY

KATEGORIE:

1. Film
 2. Serial
 3. Anime
 4. Powrót do menu głównego
 5. Zakończ program
- Wybierz opcje:

C:\Users\Pumpel\source\repos\Projekt programowanie\Debug\Projekt programowanie.exe

```
*****
**Dodajesz film do kolekcji**
*****

Podaj tytuł?: przykładowy
Podaj studio: przykładowy
Podaj gatunek/i: przykładowy
Podaj scenarzyste/tów: przykładowy
Podaj postać/cie: przykładowy
Podaj aktorów: przykładowe
Podaj reżysera: przykładowe
Podaj date premiery[xx miesiąc xxxx]: 02 styczeń 2022
Podaj średni czas trwania filmu w minutach: 211
```

C:\Users\Pumpel\source\repos\Projekt progr

```
STATYSTYKI WATCHLISTY
KATEGORIE:
1. Film
2. Serial
3. Anime
4. Cała kolekcja
5. Powrót do głównego menu
6. Zakończ program
Wybierz opcje:
```

C:\Users\Pumpel\source\repos\Projekt programowanie\Debug\Projekt program

```
Ilość obejrzanych filmów: 7
Ilość czasu zmarnowanego na filmy: 15.5 godzin.

*****

Ilość obejrzanych serialów: 3
Ilość obejrzanych odcinków serialów: 304
Ilość czasu zmarnowanego na seriale: 153 godzin.

*****

Ilość obejrzanych anime: 5
Ilość obejrzanych odcinków anime: 234
Ilość czasu zmarnowanego na anime: 92.8 godzin.
Press any key to continue . . .
```

C:\Users\Pumpel\source\repos\Projekt program

```
SPRAWDŹ SWOJĄ WIEDZĘ!!!!
MENU:
1. Stwórz pytania
2. Zagraj w quiz
3. Menu główne
4. Zakończ program
Wybierz opcje:
```

C:\Users\Pumpel\source\repos\Projekt programowanie\Debug\Projekt programow

```
*****
**Dodajesz pytania i odpowiedzi do kategorii film**
*****

Ile pytań chcesz podać: 2
Wprowadź« 1 pytanie wraz z odpowiedziami!
Podaj pytanie: przykład

Podaj odpowiedź a: przykład
Podaj odpowiedź b: przykład
Podaj odpowiedź c: przykład
Podaj odpowiedź d: przykład
Podaj poprawną odpowiedź: przykład
```


C:\Users\Pumpel\source\repos\Projekt programowanie\Debug\Projekt programowanie.exe

```
*****
**GRASZ W KATEGORI ANIME**
*****
Na ile pytań chcesz odpowiedzieć: 2

Kto jest główną postacią w Code Geass: Hangyaku o Lelouch?
A. Lelouch Lamperogue
B. Suzaku kururugi
C. Kallen Stadtfeld
D. C.C.
Twoja odpowiedź: a
Poprawna odpowiedź, zdobywasz punkt!!

Z ilu odcinków składa się anime Detah Note?
A. 27
B. 24
C. 12
D. 37
Twoja odpowiedź: d
Poprawna odpowiedź, zdobywasz punkt!!

Koniec quizu. Zdobyte punkty: 2
Press any key to continue . . .
```

C:\Users\Pumpel\source\repos\Projekt programowanie\Det

```
Usuwanie WATCHLISTY oraz QUIZÓW
KATEGORIE:
1. Film
2. Serial
3. Anime
4. Całą kolekcję
5. Pytania
6. Powrót do menu głównego
7. Zakończ program
Wybierz opcję:
```

5. Podsumowanie i wnioski.

Z postawionych założeń do projekty, wszystkie zostały zrealizowane, również udało się utworzyć możliwość zrobienia własnego quizu i zagrania w niego. Największy problem był z zapisem polskich znaków do pliku, i odczytu ich z pliku, czasem zdarza się że nadal wyskakują jakieś błędy przy kompilacji. Problem również pojawił się z brakiem należytej kontroli nad tym co użytkownik wpisuje do konsoli. Problematycznym było utworzenie quizu w ten sposób żeby użytkownik mógł grać w większą ilość pytań. Ewentualnymi ścieżkami rozwoju programy mogło by być :

- możliwość wyszukiwania konkretnych elementów kolekcji.
- usuwanie pojedynczych elementów kolekcji.
- losowa generacja pytań w quizie.
- możliwość poprawiania wprowadzanych danych.
- większa kontrola tego co wprowadza użytkownik.