Gabriel Gołda, IS 2 rok, grupa laboratoryjna 2

Projekt z przedmiotu Programowanie Obiektowe

Termin oddania projektu – 15.01.2018

Temat: Kino

1. Opis projektu.

Projekt został stworzony z myślą o obsłudze klienta kina przy wyborze filmu, który zechciałby obejrzeć. Program, który napisałem, polega na tym, że w pierwszym kroku użytkownik podaje godzinę ( obecną godzinę w której przyszedł do kina ).

Po wpisaniu godziny użytkownik zostanie poproszony o wybór jednego z trzech dostępnych gatunków filmowych. Ma on wybór między filmem akcji, fantasy lub komedią. Użytkownik wprowadza numer odpowiedni dla danego gatunku. Dla potwierdzenia wyboru wyświetla się komunikat z wybranym przez użytkownika gatunkiem.

W następnym kroku na ekranie wyświetlają się filmy, które są dostępne do obejrzenia po podanej wcześniej godzinie (tzn. jeśli użytkownik wprowadzi np. godzinę 17 to będzie miał on wybór w filmach które są dostępne po godzinie 17 a nie przed godziną 17). Docelowo w ciągu całego dnia jest możliwy wybór spośród 7 filmów. Jeśli użytkownik wprowadzi godzinę po rozpoczęciu ostatniego filmu(np. godzina 23) zostanie on poproszony o wpisanie innej godziny.   
Wszystkie filmy są pobierane z plików .txt, a więc gdy zmieni się repertuar wystarczy zmienić dany film w pliku i dalej wszystko będzie działać.

W tym kroku należy wybrać odpowiedni dla siebie film. Należy wpisać numer odpowiadający danemu filmowi. Po wyborze filmu wyświetli się komunikat z filmem który został wybrany.

Po wyborze filmu użytkownik zostanie poproszony o wpisanie 3-cyfrowego kodu rabatowego, dzięki któremu cena biletu może zostać pomniejszona o 30%. Jest 10 kodów rabatowych ( 111, 222, 333, 444, 555, 666, 777, 888, 999, 000). Po poprawnym wpisaniu jednego z nich cena zostanie pomniejszona o 30%. Zostanie wyświetlona informacja czy kod został wpisany poprawnie oraz ile wynosi cena biletu. Jeśli natomiast kod nie zostanie wpisany poprawnie lub po prostu użytkownik nie zna kodu rabatowego i pominie ten krok to cena biletu nie zmniejszy się. Zostanie wyświetlona informacja o niepoprawności kodu oraz o cenie biletu.

Tym razem nie musimy już nic wpisywać. Wyświetla się komunikat o wydrukowaniu rachunku ( zapis danych do pliku Rachunek.txt). W rachunku znajdują się informacje o wybranym gatunku, o wybranym filmie oraz cenę jaką użytkownik musi zapłacić.

Na samym końcu pojawiają się życzenia miłego seansu.

Wszystkie wpisywania danych, które użytkownik musi wprowadzić są objęte obsługą błędów, krok nie zostanie pominięty dopóki użytkownik nie wprowadzi poprawnych danych. Obsługą błędów nie jest objęte wpisywanie godziny oraz wpisywanie kodu rabatowego co moim zdaniem jest logiczne, ponieważ nie każdy użytkownik zna kod rabatowy. Jeśli byłoby to objęte obsługą błędów to musiałby on wprowadzać kod aż w końcu udałoby mu się trafić ten szczęśliwy numerek ( a ma on niecałe 10! możliwości wyboru kodu).

1. Diagram klas
2. Opis klas, metod, funkcji globalnych, użytych bibliotek

Klasa Film zawiera

* konstruktor oraz destruktor
* statyczną zmienną cena – początkowa stała cena biletu za film
* przyjaźń do klasy Plik ( bo w klasie Plik zapisujemy elementy z klasy Film)
* statyczną metodę CenaZKodemRabatowym – dzięki niej możemy uzyskać rabat 30% po wpisaniu odpowiedniego kodu
* bibliotekę iostream
* bibliotekę string – potrzebna do wpisywania tekstu do zmiennej typu string
* bibliotekę iomanip – potrzebna do ustalania pozycji tekstu
* plik nagłówkowy klasy Plik – ponieważ jest przyjaźń z klasą Plik

Klasy Akcja, Fantasy, Komedia zawierają

* konstruktor oraz destruktor
* dziedziczenie z klasy Film
* statyczną metodę CenaZKodemRabatowym – dzięki niej możemy uzyskać rabat 30% po wpisaniu odpowiedniego kodu
* bibliotekę iostream
* bibliotekę string – potrzebna do wpisywania tekstu do zmiennej typu string
* plik nagłówkowy klasy Film – ponieważ jest dziedziczenie z klasy Film

Klasa Plik zawiera

* metodę Odczyt\_danych\_z\_pliku ( z parametrami: tablica filmów, wpisana godzina, wpisany gatunek ) – pozwala ona w zależności od gatunku otworzyć odpowiedni plik z filmami które potem wybieramy do obejrzenia. Są one wszystkie zapisywane w tablicy filmów. Dodatkowo w zależności od wpisanej godziny, z tablicy filmów wyświetlają się tylko te filmy które można obejrzeć po wpisanej godzinie.
* metodę Zapis\_danych\_do\_pliku ( z parametrami: tablica filmów, wybrany numer filmu, wybrany gatunek filmu) – pozwala ona w zależności od wpisanego gatunku utworzyć plik Rachunek.txt i zapisać w nim dane dotyczące filmu tj. gatunek filmu, tytuł oraz cena za bilet.
* bibliotekę iostream
* bibliotekę string - potrzebna do wpisywania tekstu do zmiennej typu string
* bibliotekę fstream – potrzebna do pracy z plikami txt
* bibliotekę typeinfo – potrzebna aby wypisać gatunek filmu
* pliki nagłówkowe klas Akcja, Fantasy, Komedia, Film Plik – ponieważ klasa Plikwykorzystuje elementy z tych klas

Klasa Error zawiera

* konstruktor i destruktor
* metodę sprawdź\_godzinę ( z parametrem: godzina ) – obsługa błędów dotycząca tego czy użytkownik wpisał odpowiednią godzinę, tzn. między 0 a 24
* metodę sprawdź\_numer ( z parametrem: numer , godzina ) – obsługa błędów dotycząca tego czy użytkownik wpisał odpowiedni numer filmu odpowiadający jednemu z listy filmów które ma do wyboru po podanej godzinie
* metodę sprawdź\_gatunek ( z parametrem: gatunek ) – obsługa błędów dotycząca tego czy użytkownik wpisał poprawny numer odpowiadający danemu gatunkowi
* zmienną komunikat – w tej zmiennej jest tekst który pojawia się po złym wpisaniu danej liczby przez użytkownika
* bibliotekę iostream
* bibliotekę string - potrzebna do wpisywania tekstu do zmiennej typu string
* bibliotekę exception – potrzebna do obsługi błędów

1. Podsumowanie

Tematy z zajęć które zrealizowałem w projekcie:   
Pojęcie Klasy  
Konstruktor, destruktor  
Deklaracja przyjaźni  
Dziedziczenie  
Operatory wejścia/wyjścia (operacje na plikach i strumieniach wyjściowych )  
Wyjątki C++

Wnioski:

Po napisaniu tego projektu widzę w jaki sposób pisanie programów może znaleźć swoje miejsce w życiu codziennym. Właśnie stworzyłem program który mógłby być wykorzystywany w kinie. Oczywiście jest to program który wymaga jeszcze dużo dodatkowych opcji i ulepszeń aby wykorzystać go np. w automacie do kupowania biletów w kinie. Napisanie tego programu obiektowo jest bardzo funkcjonalne i ułatwia sprawę gdy chcemy np. zrobić kilka podkategorii jednej, np. dziedziczenie kilku klas z jednej. Ale wiem że dalsza nauka języka C++ pozwoli mi w przyszłości rozwinąć taki projekt, aby był bardziej funkcjonalny.