

Bartosz Żurakowski

Programowanie obiektowe – projekt

Prowadzący kurs: Radosław Idzikowski

Semestr zimowy 2019/2020

PRZEPOWIADACZ PRZYSZŁOŚCI

Program „przepowiadający przyszłość” użytkownikowi, bazujący na horoskopie i kartach tarota. Wykorzystuje metody programowania obiektowego, w programie zaimplementowane jest 6 różnych klas. Na początku program pobiera od użytkownika jego dane: imię, nazwisko, płeć oraz datę urodzenia. Są one potrzebne klasie horoskop. Następnie użytkownik dostaje do wyboru kilka funkcjonalności.

Klasy:

1. DANE

Klasa odpowiedzialna za zebranie i przechowywanie danych użytkownika. Udostępnia ona atrybuty: imię, nazwisko, płeć, dzień, miesiąc oraz rok (każdy atrybut ma też odpowiadającą mu metodą nadającą lub zwracającą jego wartość).

Są też dwa konstruktory – bezargumentowy (nadający podstawowe wartości) oraz 6-argumentowy (odpowiadające 6 atrybutom klasy, ustawia on podaną w argumencie wartość do odpowiadającego mu atrybutu za pomocą metody set...()).

W metodach jest też funkcja zaprzyjaźniona operator<<, która przeciąża operator „<<”, pozwalając wyświetlić wszystkie zapisane wartości w klasie na ekran.

2. HOROSKOP

Klasa wyznaczająca, jaki znak zodiaku itp. ma użytkownik, przypisująca mu jego wróżby, ale też zapisująca dane nowego użytkownika do klasy dane.

Klasa posiada prywatny wektor „użytkownik”, który jest typu „dane”. Do niego zapisywane są właśnie podane przez nowego użytkownika wartości.

Konstruktor bezargumentowy- otwiera plik z danymi użytkowników i zapisuje je do wektora „użytkownik”.

Funkcja nowyUzytkownik – otwiera plik z danymi użytkowników i zapisuje do niego wprowadzane wartości, każda jest zabezpieczona przed błędnym wprowadzeniem (np. sprawdza czy wiek jest poprawny). Następnie wprowadzone dane zapisuje do wektora „użytkownik”.

Funkcja admin – po podaniu poprawnego hasła wyświetla dane osób, które do tej pory skorzystały z programu.

W dokumentacji początkowej admin miał być klasą, jednak nie było sensu tworzyć do tego osobnej klasy, nie posiadała by ona żadnych sensownych atrybutów i tylko jedną metodę wyświetlającą dane, więc została ona tylko funkcją w klasie horoskop.

3. LOSOWA

Klasa wyznaczająca użytkownikowi losową wróżbę, ale też pozwalająca na sprawdzenie swojego szczęścia poprzez rzut kośćmi (rzut kośćmi początkową miał być osobną klasą, lecz jak wyżej- nie było sensu).

Posiada prywatny wektor typu string „losowe”, do którego zapisywane są z pliku wróżby, z których później będzie losowanie tej jednej wyświetlonej w programie.

Dwie metody:

Losuj- losująca wróżbę, oraz

rzutKosciami- symulująca dwa rzuty kośćmi, na podstawie zliczonych oczek z obu rzutów wyświetla informację o dzisiejszym szczęściu „rzucającego”.

4. MALE

Klasa reprezentująca jeden z dwóch rodzajów kart w tarocie- karty małe arkana. Tarot składa się z małych i wielkich arkana.

Posiada trzy chronione (protected) atrybuty: nazwa (określająca czy jest to karta buław, kielichów, mieczy czy monet), numer (określający jaką to karta- od 1 do 10 lub as, rycerz, król, królowa, giermek) oraz znaczenie (zawierający znaczenie każdej pojedynczej karty).

Atrybuty są protected ponieważ późniejsza klasa Duze dziedziczy po klasie Male.

Klasa posiada konstruktor bezargumentowy oraz 3-argumentowy, metody ustalające i zwracające wartość dla danego atrybutu, oraz metodę operator<< (przeciążony operator <<) pozwalająca wyświetlić, jaką kartę dostał użytkownik oraz informację o niej.

5. DUZE

Klasa reprezentująca karty wielkie arkana, dziedzicząca po klasie Male.

Rozszerzona o następujące atrybuty i metody: numerArkanu (są 22 karty wielkie arkana, każda ma numer, który jest tu zapisywany, osobne co do klasy Male, ponieważ tu jest zapisana liczba w systemie dziesiętnym, a w klasie małe słownie), znakAstrologiczny, atrybuty (każda karta ma swój atrybut, np. przygoda), miłosc (zapisywane są tu wróżby co do miłości, które określa karta wielkie arkana), finanse (analogicznie jak w przypadku miłosc).

Dwa konstruktory- bezargumentowy i 7 argumentowy (5 z tej klasy i dwa dziedziczone) oraz metody ustalające i zwracające wartość dla każdego atrybutu.

W tej klasie również zastosowany jest przeciążony operator <<, który pozwala wyświetlić nam jaką kartę dostał użytkownik oraz informację o niej.

6. KARTA

Klasa, która łączy klasy Male i Duze tworząc talię kart tarota, co pozwala wybrać karty dla użytkownika.

Dwa prywatne wektory talia1 i talia2- odpowiednio typu Duze i Male. Ponieważ w rozkładzie tarota ma znaczenie, czy i na którym miejscu będzie to karta wielkich czy małych arkana, dlatego nie zostaną one połączone w jedną spójną talię, ponieważ wtedy byłaby losowość wyboru- a pierwsza w wybranym przeze mnie rozkładzie musi być karta wielka, następnie 3 małe.

Metody:

wczytajDuze- otwiera plik, w którym zapisane są wielkie karty arkana z ich właściwościami, następnie zapisuje wszystkie karty do wektora talia1.

wyswietlDuze- wyświetla wszystkie karty zapisane w talia1.

wczytajMale- analogicznie do wczytajDuze.

wyswietlMale- analogicznie do wyswietlDuze.

losujDuza- losuje dla użytkownika jedną kartkę wielkich arkana i wyświetla informację o niej.

losujMala- analogicznie.

losujMilosc- losuje kartę wielkich arkana i wyświetla przypisaną do niej wróżbę co do miłości za pomocą funkcji getMilosc z klasy Duze.

losujFinanse- losuje kartę wielkich arkana i wyświetla przypisaną do niej wróżbę co do finansów za pomocą funkcji getFinanse z klasy Duze.

W programie jest też plik nagłówkowy „libs.h” zawierający biblioteki potrzebne do działania programu, dołączany do każdej klasy, zabezpieczony dyrektywą #pragma once.

Porównując do dokumentacji początkowej, zrezygnowałem z klas admin, losowa wróżba, szczęście w miłości oraz bogactwo- nie było sensu tworzyć z nich osobnych klas, zostały one zawarte jako metoda lub zbiór metod w innych istniejących klasach. Był też zapis o możliwej klasie menu- ona również nie powstała, menu programu zostało zaimplementowane w funkcji main.

Hierarchia klas w programie:

