Grzegorz Aleksiuk Warszawa 14.06.2018

Dokumentacja

Projekt 3 – *,,System zarządzania pracownikami”*

**Opis programu**

Program służy do zarządzania pracownikami w firmie, a także do monitorowania stanu firmy. Firma jest wczytywana do programu z pliku .txt o dowolnej nazwie. Plik z firmą musi być odpowiednio spreparowany, jeśli plik będzie zawierał błąd program nie będzie prawidłowo działać. Program wykorzystuje klasę Company, która znajduje się w pliku nagłówkowym Company.h i klasy Date z pliku nagłówkowego Calendar.h.

**Funkcje w pliku main.cpp**

***void wczytaj\_firme(string nazwa);*** -wczytuje plik o podanej nazwie

***void menu();*** -rozpoczyna działanie systemu zarządzania

***void system();*** -główne menu zarządzania systemem

***void system\_dzial(int x);*** -podmenu zarządzania działem

***void system\_pracownik(int x, int y);*** -podmenu zarzadzania pracownikiem

***void lista\_dzialow();*** -wypisuje listę działów w firmie

***void wypisz\_raport();*** -wypisuje aktualny stan firmy

***void wczytaj\_pracownika(int x);*** -wczytuje nowego pracownika i dodaje go do działu

***void zapisz\_raport();*** -zapisuje raport o stanie firmy w pliku .txt

***void zapisz\_firme(string nazwa);*** -zapisuje firmę w podanym pliku

***int sprawdz(int a, int b, int z);*** -sprawdza czy użytkownik wpisał odpowiednia opcje z menu

**Informacje o klasach**

**Klasa Person**

Przechowuje informacje dotyczące osoby: imię, nazwisko, płeć i wiek.

**Klasa Work**

Przechowuje informacje o pracy: nazwa stanowiska, pozycja, wypłatę i dochód który generuje dla firmy.

**Metody:**

***string show();*** - zwraca string z informacjami o pracy

***int get\_income();*** - zwraca generowany przychód

***int get\_salary();*** - zwraca wypłatę

***void change\_job\_title(string);*** - zmienia stanowisko

***void change\_position();*** - zmienia pozycje na jedną wyżej ale nie wyżej niż kierownik

***void change\_income(int);*** - zmienia generowany przychód

***void change\_salary(int);*** - zmienia wypłatę

***void promotion(string, int);*** - awansuje

***void rise(int);*** - zmienia wypłatę

**Klasa Employee**

Dziedziczy po klasach Person i Work

**Klasa Area**

Klasa służąca tylko do porozumiewania się z innymi obiektami.

**Metody wirtualne**

***virtual int get\_size() ;***

***virtual int get\_income();***

***virtual int get\_outflow();***

**Klasa Branch**

Dziedziczy po klasie Area. Przetrzymuje nazwę działu i vector z pracownikami.

**Metody**

***void add\_worker(Employee);*** - dodaje do vectora pracownika

***void fire\_worker(int);***  - usuwa pracownika z vectora

***void promote\_worker(int , string, int);*** - awansuje podanego pracownika

***void give\_rise(int, int);***  -podwyższa płacę dla pracownika

***virtual int get\_inome();*** -zwraca łączną sumę przychodów generowanych przez pracowników w tym dziale

***virtual int get\_outflow();*** - zwraca łączną sumę wypłat pracowników

***virtual int get\_size();*** - zwraca ilość pracowników w tym dziale

***string get\_name();*** - zwraca nazwę działu

**Klasa Company**

Przetrzymuje nazwę firmy, ilość pieniędzy posiadaną przez firmę, dochód, koszty i vector z klasami Branch. Dziedziczy po klasie Area

**Metody**

***void add\_branch(Branch);*** - dodaje dział do vectora

***void add\_worker(int, Employee);*** - dodaje pracownika do podanego działu

***void promote\_worker(int, int, string, int);*** - daje awans dla podanego pracownika w danym dziale

***void fire\_worker(int, int);*** - zwalnia podanego pracownika

***void rise\_worker(int, int, int);*** - daje podwyżkę dla podanego pracownika o daną kwotę

***int show\_money();*** - zwraca ilość pieniędzy posiadaną przez firmę

***virtual int get\_income();*** – zwraca przychód całej firmy

***virtual int get\_outflow();*** – zwraca wszystkie wydatki firmy

***virtual int get\_size();***  - zwraca liczbę działów w firmie

***int get\_workers\_size();***  - zwraca ilość pracowników w firmie

**Klasa Time**

Klasa służy do komunikacji z innymi obiektami

**Klasa Date**

Dziedziczy po Klasie Time. Przechowuje dzień, miesiąc i rok;

**Metody**

***string get\_date();*** - zwraca string z aktualną datą

***int get\_day();*** - zwraca dzień

***int get\_month();*** - zwraca miesiąc

***int get\_year();***  - zwraca rok

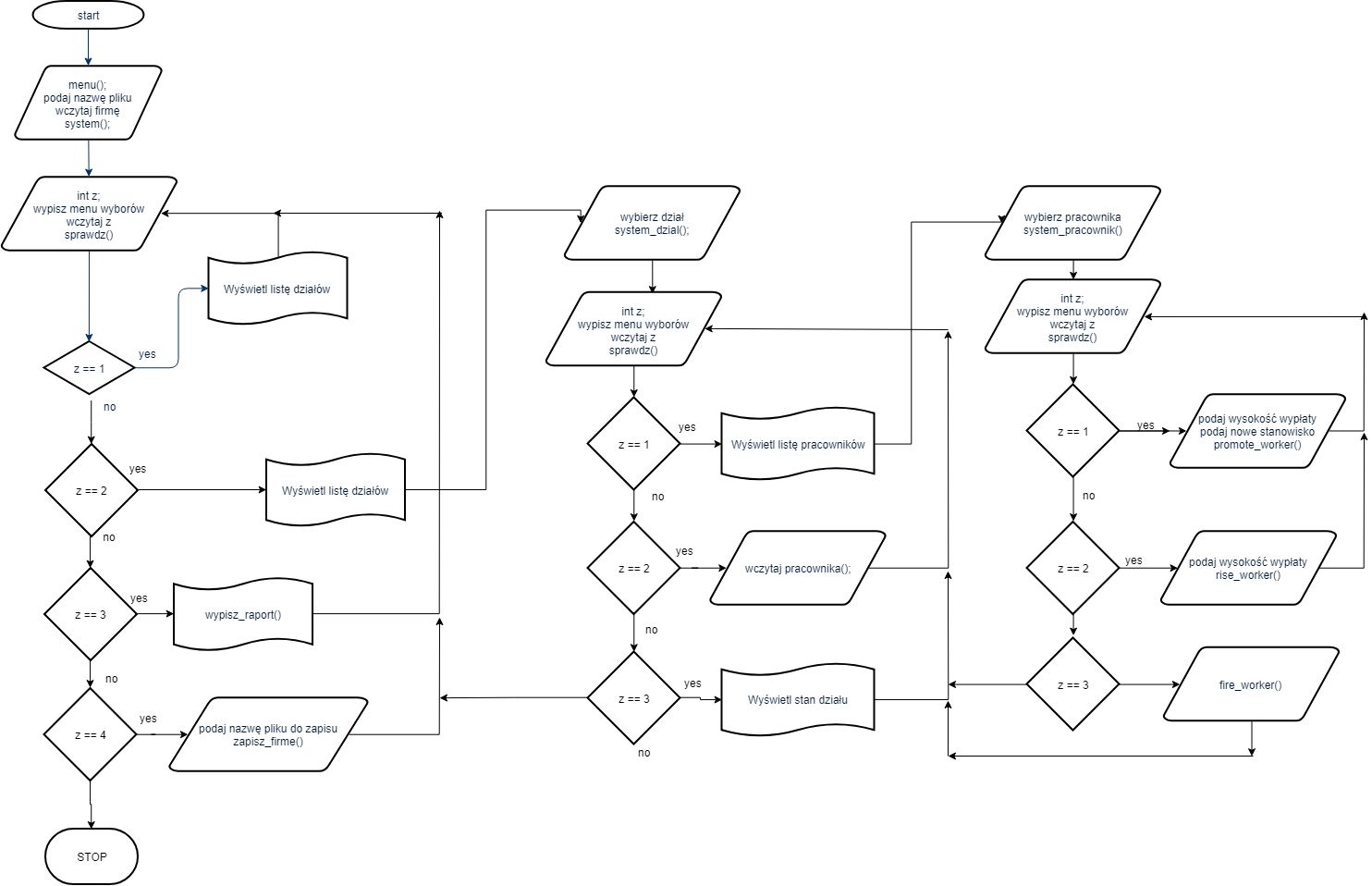
***void next\_day();*** - przechdzi na następny dzień

***void next\_week();*** - przechodzi o 7 dni do przodu

***virtual string show()*** – wypisuje dzisiejszą datę

**Zapis**

Program ma możliwość zapisywania raportów ze stanem firmy w plikach o nazwie Raport[dd.mm.yyyy].txt a także zapisywać stan firmy w pliku .txt o podanej przez użytkownika nazwie.

**Schemat działania systemu zarządzania pracownikami**

**Plik z firmą**

Musi być w formacie .txt i musi także być odpowiednio spreparowany, w innym przypadku program nie będzie działać poprawnie.

**Schemat**

[DATA] [dd.mm.yyyy]

[NAZWA FIRMY]

[PIENIADZE FIRMY]

[ILOŚĆ DZIAŁÓW]

[nr działu. ] [NAZWA DZIALU]

[LICZBA PRACOWNIKÓW DZIAŁU]

[nr pracownika. ] [IMIE] [NAZWISKO] [płeć] [STANOWISKO] [pozycja] [WYPŁATA] [PRZYCHÓD]

[płeć] = mezczyzna/kobieta

[pozycja] = prezes/kierownik/pracownik/stazysta/praktykant

**Przykład**

14.06.2018

Multi Packaging Solutions

20000000

3

1. HR

2

1. Lucjan Czarnecki mezczyzna 30 Rozwoj kierownik 7000 10000

2. Jowita Tomaszewska kobieta 48 Szkoleniowiec pracownik 5000 6000

2. IT

2

1. Rafal Majewski mezczyzna 40 Senior kierownik 9000 12000

2. Szczepan Wysocki mezczyzna 35 Developer pracownik 7000 9000

3. PR

1

1. Augustyn Jasinski mezczyzna 42 Menager kierownik 8000 8000