Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний

інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 3 з дисципліни

«Основи програмування 2. Модульне програмування»

Варіант 25

Виконав студент ІП-14 Радзівіло Валерія Артемівна

(шифр, прізвище, ім'я, по батькові)

Перевірив

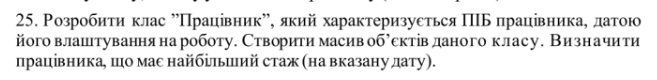
( прізвище, ім'я, по батькові)

Київ 2022

**Лабораторна робота №3**

**Тема роботи:** Класи та об’єкти

**Мета роботи:** Вивчити механізми створення і використання класів та об’єктів.



**Вирішення задачі С++:**

**main:**

#include <iostream>  
#include "header.cpp"  
using namespace std;  
int main() {  
 int amount;  
 get\_amount(amount);  
 Worker\*pers[amount];//making array of Workers  
 string worker\_most;  
 int worked[amount];  
 int max\_work;  
 string user\_date = user\_year();  
 for(int i =0; i < amount; i++)  
 {  
  
 string PIB, date;  
 make\_worker\_list(PIB, date);  
 Worker worker\_pers(PIB, date);  
 pers[i] = &worker\_pers; //write an object int array  
 pers[i]->print();  
 worked[i] = pers[i]->worked\_for(date, user\_date);  
 cout << "Worked till the date for: " << worked[i] << endl;  
 max\_work = worked[0];  
 if( worked[i] >= max\_work) {  
 max\_work = worked[i];  
 worker\_most = PIB;  
 }  
 }  
 cout << endl << "THE RESULT" << endl;  
 cout <<" Worked the most: " << worker\_most << endl;  
 cout <<"He/she worked for: " <<max\_work << " days"<< endl;  
 return 0;  
}

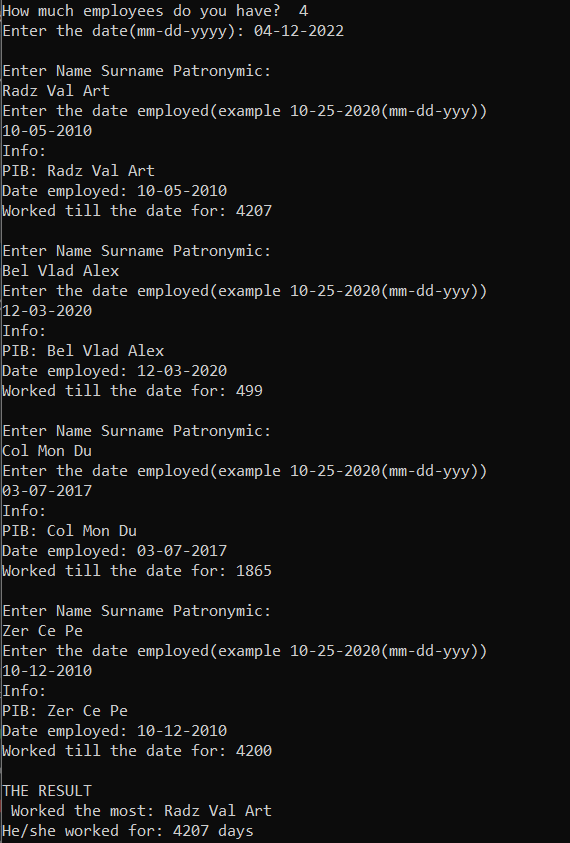
**header.h:**

//  
// Created by leraz on 28-03-22.  
//  
  
#ifndef CPP\_HEADER\_H  
#define CPP\_HEADER\_H  
#include <iostream>  
using namespace std;  
class Worker{  
 string PIB;  
 string date\_employed;  
public:  
 Worker(string x,string y)  
 {  
 PIB = x;  
 date\_employed = y;  
 };  
 void print();  
 int worked\_for(string date\_employed, string user\_date);  
  
};  
void find\_worker(string user\_date);  
  
string user\_year();  
void make\_worker\_list(string &PIB, string &date\_employed);  
int date\_difference(string date\_one, string date\_two);  
#endif //CPP\_HEADER\_H

**header.cpp:**

#include "header.h"  
  
void get\_amount(int &amount)  
{  
 cout <<"How much employees do you have? ";  
 cin >> amount;  
}  
  
void make\_worker\_list(string &PIB, string &date\_employed)  
{  
 cin.ignore();  
 cout << "Enter Name Surname Patronymic: "<< endl;  
 getline(cin, PIB);  
 cout << "Enter the date employed(example 10-25-2020(mm-dd-yyy)) "<< endl;  
 getline(cin, date\_employed);  
}  
  
string user\_year() {  
 cout << "Enter the date(mm-dd-yyyy): ";  
 string date;  
 cin.ignore();  
 getline(cin, date);  
 return date;  
}  
  
int Worker::worked\_for(string date\_employed, string user\_date) {  
 int worked = date\_difference(user\_date, date\_employed);  
 return abs(worked);  
}  
  
  
int date\_difference(string date\_one, string date\_two)  
{  
 int month\_one = stoi(date\_one.substr(0,2));  
 int day\_one = stoi(date\_one.substr(3,2));  
 int year\_one = stoi(date\_one.substr(6,4));  
 int month\_two = stoi(date\_two.substr(0,2));  
 int day\_two = stoi(date\_two.substr(3,2));  
 int year\_two = stoi(date\_two.substr(6,4));  
  
 //calc  
 int year\_dif;  
 int month\_dif;  
 int day\_dif;  
 year\_dif=year\_two-year\_one;  
 month\_dif=month\_two-month\_one;  
 day\_dif=day\_two-day\_one;  
  
 //case if dd is negative i.e. current dd is less than first date  
 if(day\_dif<0)  
 {  
 month\_dif--;  
 day\_dif+=30;  
 }  
  
 //case if mm is negative i.e. current mm is less than first date  
 if (month\_dif<0)  
 {  
 year\_dif--;  
 month\_dif+=12;  
 }  
 int result;  
 result = year\_dif \* 365 + month\_dif \* 30 + day\_dif;  
 return result;  
}  
  
void Worker::print() {  
 cout << "Info: "<< endl;  
 cout << "PIB: " << PIB << endl;  
 cout << "Date employed: "<< date\_employed << endl;  
}

**Результат:**



**Висновок**: У цій лабораторній роботі було вивчено механізми створення і використання класів та об’єктів. Була постановлена задача, в якій визначався список працівників, введений користувачем. У алгоритмі були уточнені обмеження з використанням умов при яких задача буде виконана правильно. За допомоги цієї лабораторної роботи можливо складання списку працівників та розрахунку який працівник пропрацював більше за інших.