

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»

«Класи та об'єкти»

Варіант 22

Виконав студент _____ ІП-15_Мешков_Андрій_Ігорович_____

(шифр, прізвище, ім'я, по батькові)

Перевірів _____ Вєчерковська Анастасія Сергіївна _____

(прізвище, ім'я, по батькові)

Київ 2022

Лабораторна робота 3

Класи та об'єкти

Мета – вивчити механізми створення і використання класів та об'єктів.

Варіант 22

Завдання.

22. Розробити клас "Студент", який характеризується ПІБ студента, номером його групи, датою народження (у форматі ММ-ДД-РРРР). Створити масив об'єктів даного класу. Визначити найстаршого студента вказаної групи (на вказану дату).

Код C++

lab1.cpp

```
1  #include "function.hpp"
2  #include <iostream>
3  #include <string>
4
5  using namespace std;
6
7
8
9  int main() {
10     setlocale (LC_ALL, "Russian");
11     int num_st;
12     int old_number;
13     char flag = 'y';
14     cout<<"Введіть кількість студентів: ";
15     cin>>num_st;
16     cout<<endl;
17
18     Student *catalog = new Student [num_st]; //масив студентів
19     catalog = array_objects(catalog, num_st); //заповнення масиву студентів
20     while(flag == 'y'){
21         old_number = oldest(catalog, num_st); //знаходження найстаршого студента
22         cout<<catalog[old_number].name<<endl;
23         cout<<"\nХотите еще найти студента? (y/n): ";
24         cin>>flag;
25     }
26
27     delete [] catalog; //видалення масиву
28
29 }
30 |
```

```
1  #ifndef function_hpp
2  #define function_hpp
3  #include <string>
4
5  #include <stdio.h>
6
7  using namespace std;
8
9  class Student{//клас Студент
10 public:
11     Student() {}
12     Student(string name, string number, string dOb){
13         this->name = name;
14         this->number = number;
15         this->dOb = dOb;
16     }
17     string name;//ім'я
18     string number;//номер групи
19     string dOb;//дата народження
20 };
21
22 Student* array_objects(Student*, int);
23 string verify_n(string);
24 bool is_Liter(string);
25 bool three(string);
26 string verify_gr(string);
27 bool upper_dig(string);
28 string verify_d(string);
29 bool time_d(string);
30 int oldest(Student*, int);
31 bool who_older(string, string);
32
33 #endif
34
```

function.cpp

```
1  #include "function.hpp"
2  #include <iostream>
3  #include <string>
4
5  using namespace std;
6
7  Student* array_objects(Student* catalog, int num_st){
8      setlocale (LC_ALL, "Russian");
9      string name_st, number, dOb;
10     for(int i=0; i<num_st; i++){
11         cin.ignore();
12         cout<<"Введите ваше ФИО: ";
13         getline(cin, name_st);
14         number = verify_n(name_st);//перевірка вводу
15         cout<<"Введите номер вашей группы(в формате ББ-ЦЦ): ";
16         getline(cin, number);
17         number = verify_gr(number);//перевірка вводу
18         cout<<"Введите вашу дату рождения(в формате ММ-ДД-ГГГГ): ";
19         getline(cin, dOb);
20         dOb = verify_d(dOb);//перевірка вводу
21
22         catalog[i] = Student(name_st, number, dOb);//запис об'єкту у масив
23     }
24     cout<<endl;
25     return catalog;
26 }
27
28 string verify_n(string name){
29     bool flag_not_lit = is_Liter(name);//перевірка, що ім'я має тільки літери, виключно дефіс
30     bool three_word = three(name);//перевірка, що має три слова роздільні пробілами
31
32     while(name.size()>50 || flag_not_lit || three_word){
33         cin.ignore();
34         cout << "Введите еще раз: ";
35         getline(cin, name);
36         flag_not_lit = is_Liter(name);
37         three_word = three(name);
38     }
39     return name;
40 }
41 }
42
```

```

43 bool is_liter(string name){
44     bool flag = false;
45     for(int i=0; i<name.size(); i++){
46         if(name[i] > 32 && name[i] < 45 || name[i] > 45 && name[i] < 65){//перевірка на літери та дефіс
47             flag = true;
48             break;
49         }
50     }
51     return flag;
52 }
53
54 bool three(string name){
55     bool flag = false;
56     int x = count(name.begin(), name.end(), ' ');//шукаємо два пробіли 2 ⚠ Implicit conversion loses integer preci
57     if(x != 2){
58         flag = true;
59     }
60     return flag;
61 }
62
63 string verify_gr(string number){
64     bool flag_liter_dig = upper_dig(number);//перевірка формату
65
66     while(number.size()!=5 || number[2]!='-' || flag_liter_dig){
67         cin.ignore();
68         cout << "Введіть ще раз: ";
69         getline(cin, number);
70         flag_liter_dig = upper_dig(number);
71     }
72     return number;
73 }
74
75 bool upper_dig(string num){
76     bool flag = false;
77     //перші два символи - великі латинські, два останні - цифри
78     if(!isupper(num[0]) || !isupper(num[1]) || !isdigit(num[3]) || !isdigit(num[4])){
79         flag = true;
80     }
81     return flag;
82 }
83

```

```

84 string verify_d(string date){
85     bool flag_time = time_d(date); //перевірка дати
86
87     while(date.size()!=10 || date[2]!='-' || date[5]!='-' || flag_time){
88         cin.ignore();
89         cout << "Введіть ще раз: ";
90         getline(cin, date);
91         flag_time = time_d(date);
92     }
93     return date;
94 }
95
96 bool time_d(string date){
97     bool flag = true;
98     int m, d, y;
99     m = atoi(date.substr(0, 2).c_str());
100    d = atoi(date.substr(3, 2).c_str());
101    y = atoi(date.substr(6, 4).c_str());
102    if(y>=1900 && y<=2019){ //рік народження з 1900 до 2019
103        if(m>=1 && m<=12){ //перевірка на місяці
104            if(m == 1 || m == 3 || m == 5 || m == 7 || m == 8 || m == 10 || m == 12){ //місяці з 31 днем
105                if(d>=1 && d<=31){
106                    flag = false;
107                }
108            }
109            if(m == 4 || m == 6 || m == 9 || m == 11){
110                if(d>=1 && d<=30){ //місяці з 30 днями
111                    flag = false;
112                }
113            }
114            else{ //лютий у звичайний та високосний рік
115                if(y%4 == 0){
116                    if(d>=1 && d<=29){
117                        flag = false;
118                    }
119                }
120                else{
121                    if(d>=1 && d<=28){
122                        flag = false;
123                    }
124                }
125            }
126        }
127    }
128    return flag;
129 }
130

```

```

131 int oldest(Student* catalog, int num){
132     int n_st;
133     string oldest="01-01-2020";
134     bool older;
135     string group;
136     cout<<"В якій групі знайти старшого студента? ";
137     cin>>group;
138     for(int i=0; i<num; i++){
139         if(catalog[i].number == group){
140             older = who_older(catalog[i].dOb, oldest); //порівнюємо вік
141             if(older){
142                 oldest = catalog[i].dOb;
143                 n_st = i;
144             }
145         }
146     }
147     return n_st;
148 }
149
150 bool who_older(string date, string oldest){
151     bool flag = false;
152     int m, d, y, m_o, d_o, y_o;
153     m = atoi(date.substr(0, 2).c_str());
154     d = atoi(date.substr(3, 2).c_str());
155     y = atoi(date.substr(6, 4).c_str());
156     m_o = atoi(oldest.substr(0, 2).c_str());
157     d_o = atoi(oldest.substr(3, 2).c_str());
158     y_o = atoi(oldest.substr(6, 4).c_str());
159     if(y<y_o){ //порівняння років
160         flag = true;
161     }
162     if(y==y_o){
163         if(m<m_o){ //порівняння місяців
164             flag = true;
165         }
166         if(m==m_o){
167             if(d<d_o){ //порівняння днів
168                 flag = true;
169             }
170             if(d==d_o){
171                 if(oldest == "01-01-2020"){
172                     flag = true;
173                 }
174             }
175         }
176     }
177     return flag;
178 }

```



Using the result



Using the result



Using the result



Using the result



Using the result



Using the result

Результат

C++

Введите количество студентов: 3

Введите ваше ФИО: Meshkov Andrii Igorovich

Введите номер вашей группы(в формате ББ-ЦЦ): IP-15

Введите вашу дату рождения(в формате ММ-ДД-ГГГГ): 21-05-2004

Введите еще раз: 05-21-2004

Введите ваше ФИО: Noviy Ivan Petrovich

Введите номер вашей группы(в формате ББ-ЦЦ): IP-15

Введите вашу дату рождения(в формате ММ-ДД-ГГГГ): 03-12-2003

Введите ваше ФИО: Torba Irina Ivanovna

Введите номер вашей группы(в формате ББ-ЦЦ): IP-13

Введите вашу дату рождения(в формате ММ-ДД-ГГГГ): 01-01-2000

В какой группе найти старшего студента? IP-15

Noviy Ivan Petrovich

Хотите еще найти студента? (y/n): y

В какой группе найти старшего студента? IP-13

Torba Irina Ivanovna

Хотите еще найти студента? (y/n): n

Program ended with exit code: 0