

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 № 2 з дисципліни
«Основи програмування 2. Методології програмування»
«Файли даних. Бінарні файли»

Варіант __16__

Виконав студент __ІП-15, Куманецька Ірина Вікторівна__
Перевірив __Вечерковська Анастасія Сергіївна__

Київ 2022

Лабораторна робота 2

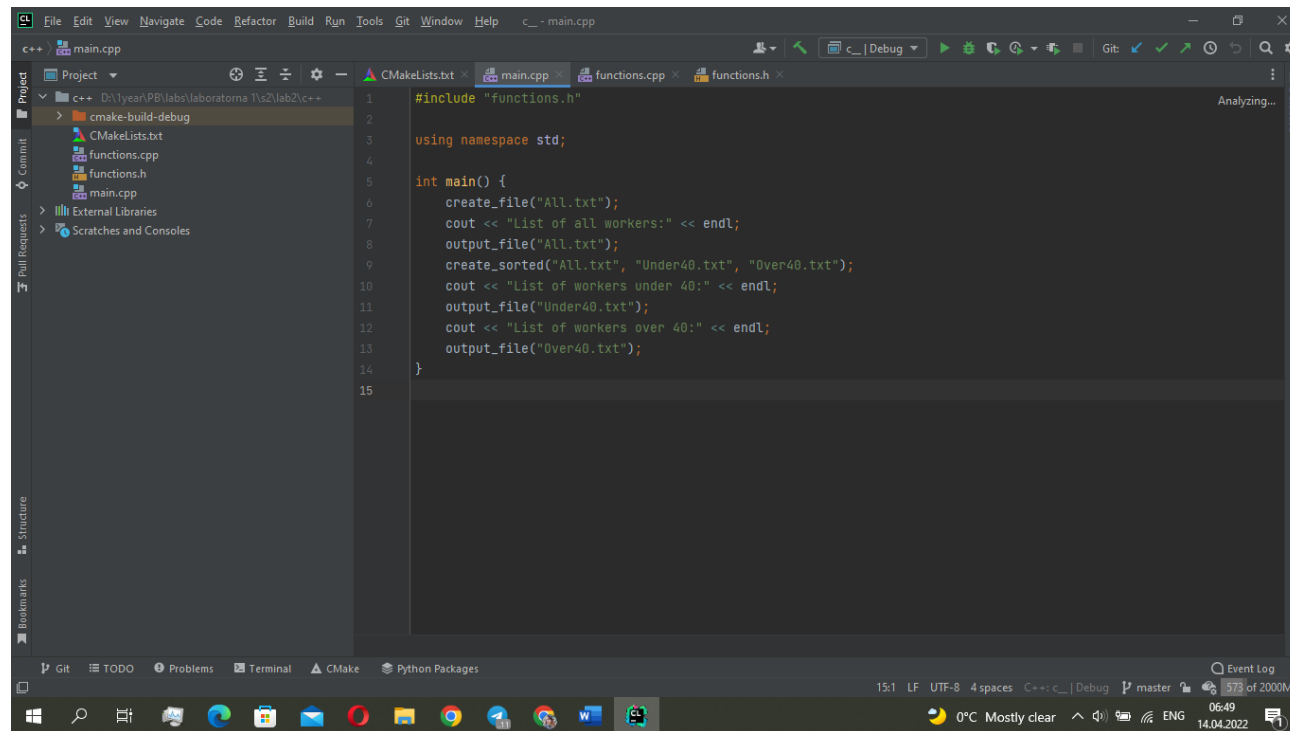
Мета – вивчити особливості створення і обробки бінарних файлів даних.

Індивідуальне завдання

Варіант 16

16. Створити файл із списком співробітників підприємства: ПІБ працівника, його дата народження (ДД.ММ.РРРР), табельний номер, стать (чоловік / жінка). При додаванні даних виконувати перевірку на вік: він повинен бути не менше 20-ти та не більше 60-ти років. Створити два нових файли: перший містить список працівників, яким не більше 40 років, другий – список інших працівників.

Виконання на C++



```
1 #include "functions.h"
2
3 using namespace std;
4
5 int main() {
6     create_file("All.txt");
7     cout << "List of all workers:" << endl;
8     output_file("All.txt");
9     create_sorted("All.txt", "Under40.txt", "Over40.txt");
10    cout << "List of workers under 40:" << endl;
11    output_file("Under40.txt");
12    cout << "List of workers over 40:" << endl;
13    output_file("Over40.txt");
14 }
15
```

This screenshot shows the Visual Studio Code editor with the `functions.h` file open. The file contains the following code:

```
1 #include <iostream>
2 #include <string>
3 #include <fstream>
4 #include <cmath>
5 #include <iomanip>
6 #include <stdio>
7
8 using namespace std;
9
10 struct Date;
11 struct Worker;
12 string get_number();
13 void create_file(const string& name);
14 Worker input_data();
15 string get_gender();
16 void output_file(const string& name);
17 bool is_number(const string& word);
18 string get_day();
19 string get_month();
20 int get_year();
21 string get_name();
22 int get_amount();
23 void create_sorted(const string& old, const string& under, const string& over);
24
```

The interface includes a sidebar with a project tree, a top menu bar, and a bottom status bar showing the current file, line numbers, and system information.

This screenshot shows the Visual Studio Code editor with the `functions.cpp` file open. The file contains the following code:

```
1 #include "functions.h"
2
3 using namespace std;
4
5 struct Date{
6     char day[4];
7     char month[4];
8     int year;
9 };
10
11 struct Worker{
12     char name[21];
13     char number[9];
14     char gender[7];
15     Date birth;
16 };
17
18 void create_file(const string& name){
19     ofstream file;
20     string ch;
21     cout << "Would you like to rewrite information (Y/N)? ";
22     cin >> ch;
23     while (ch != "Y" and ch != "y" and ch != "N" and ch != "n") {
24         cout << "Wrong input. Enter 'Y' or 'N'. ";
25     }
26     create_file
27 }
```

The interface includes a sidebar with a project tree, a top menu bar, and a bottom status bar showing the current file, line numbers, and system information.

This screenshot shows the Visual Studio Code editor with the 'functions.cpp' file open. The 'create_file' function is visible, which prompts the user to create or overwrite a file. It then reads the number of workers to add and iterates through that number, prompting for worker data (name and birth year) and writing it to the file. The function also includes a validation loop for the birth year, ensuring it is between 1902 and 1962.

```
17
18 void create_file(const string& name){
19     ofstream file;
20     string ch;
21     cout << "Would you like to rewrite information (Y/N)? ";
22     cin >> ch;
23     while (ch != "Y" and ch != "y" and ch != "N" and ch != "n") {
24         cout << "Wrong input. Enter 'Y' or 'N'. ";
25         cin >> ch;
26     }
27     if (ch == "Y" or ch == "y") file.open( s name, mode ios::binary);
28     else file.open( s name, mode ios::binary | ios::app);
29     int num = get_amount();
30     for (int i = 0; i < num; ++i) {
31         Worker person = input_data();
32         while (person.birth.year > 2002 or person.birth.year < 1962){
33             cout << "Age of this person is out of range. Try again." << endl;
34             person = input_data();
35         }
36         file.write( s (char*)&person, n sizeof(Worker));
37         cout << "Recorded" << endl;
38     }
39     file.close();
40 }
41
42 void create_sorted(const string& old, const string& under, const string& over){
43     ifstream file( s old, mode ios::binary);
44     get_amount
```

This screenshot shows the Visual Studio Code editor with the 'functions.cpp' file open. The 'create_sorted' function is visible, which reads the number of workers to add and iterates through that number, prompting for worker data (name and birth year) and writing it to the file. The function also includes a validation loop for the birth year, ensuring it is between 1902 and 1962. The 'get_amount' function is also visible, which prompts the user to enter the number of workers to add and returns that number.

```
41
42 void create_sorted(const string& old, const string& under, const string& over){
43     ifstream file( s old, mode ios::binary);
44     ofstream un( s under, mode ios::binary), ov( s over, mode ios::binary);
45     Worker person{};
46     while (file.read( s (char*)&person, n sizeof(Worker))){
47         if (person.birth.year <= 1982) ov.write( s (char*)&person, n sizeof(Worker));
48         else un.write( s (char*)&person, n sizeof(Worker));
49     }
50     file.close();
51     un.close();
52     ov.close();
53 }
54
55 int get_amount(){
56     string num;
57     cout << "Enter number of workers to add: ";
58     cin >> num;
59     cin.ignore();
60     while (!is_number( word: num) or stoi( str: num) < 1){
61         cout << "Enter a positive integer: ";
62         cin >> num;
63         cin.ignore();
64     }
65     return stoi( str: num);
66 }
67
input_data
```

This screenshot shows the Visual Studio Code editor with the `functions.cpp` file open. The code defines a `Worker` struct and a `get_name()` function. The `Worker` struct has fields for `name`, `number`, `gender`, `birth.day`, `birth.month`, and `birth.year`. The `get_name()` function prompts the user to enter a full name (up to 20 symbols) and validates its length. If the name is too long, it prompts the user to try again.

```
65 return stoi(str_num);
66
67
68 Worker input_data(){
69     Worker per{};
70     strcpy( Dest: per.name, Source: get_name().c_str());
71     strcpy( Dest: per.number, Source: get_number().c_str());
72     strcpy( Dest: per.gender, Source: get_gender().c_str());
73     strcpy( Dest: per.birth.day, Source: get_day().c_str());
74     strcpy( Dest: per.birth.month, Source: get_month().c_str());
75     per.birth.year = get_year();
76     return per;
77 }
78
79 string get_name(){
80     string name;
81     cout << "Full name (up to 20 symbols): ";
82     getline( &cin, &name);
83     while(name.length() > 20){
84         cout << "Too Long. Try again: ";
85         getline( &cin, &name);
86     }
87     return name;
88 }
89
90 string get_gender(){
91     cout << "Gender (M/F): ";
92     get_gender
```

This screenshot shows the Visual Studio Code editor with the `functions.cpp` file open, displaying the implementation of the `get_gender()` and `get_day()` functions. The `get_gender()` function prompts the user to enter their gender (M or F) and validates the input. The `get_day()` function prompts the user to enter the day of birth (1 to 31) and validates the input.

```
90 string get_gender(){
91     cout << "Gender (M/F): ";
92     string gen;
93     cin >> gen;
94     cin.ignore();
95     while (not(gen == "M" or gen == "m" or gen == "F" or gen == "f")) {
96         cout << "Wrong input. Enter letter M for male or F for female: ";
97         cin >> gen;
98         cin.ignore();
99     }
100     if (gen == "M" or gen == "m") return "male";
101     else if (gen == "F" or gen == "f") return "female";
102 }
103
104 string get_day(){
105     string d;
106     cout << "Day of birth: ";
107     cin >> d;
108     cin.ignore();
109     while (!is_number( word: d) or stoi( int: d) < 1 or stoi( int: d) > 31){
110         cout << "Wrong input. Enter a number from 1 to 31: ";
111         cin >> d;
112         cin.ignore();
113     }
114     if (stoi( int: d) < 10) d = '0' + d;
115     return d+'.';
116 }
117
```

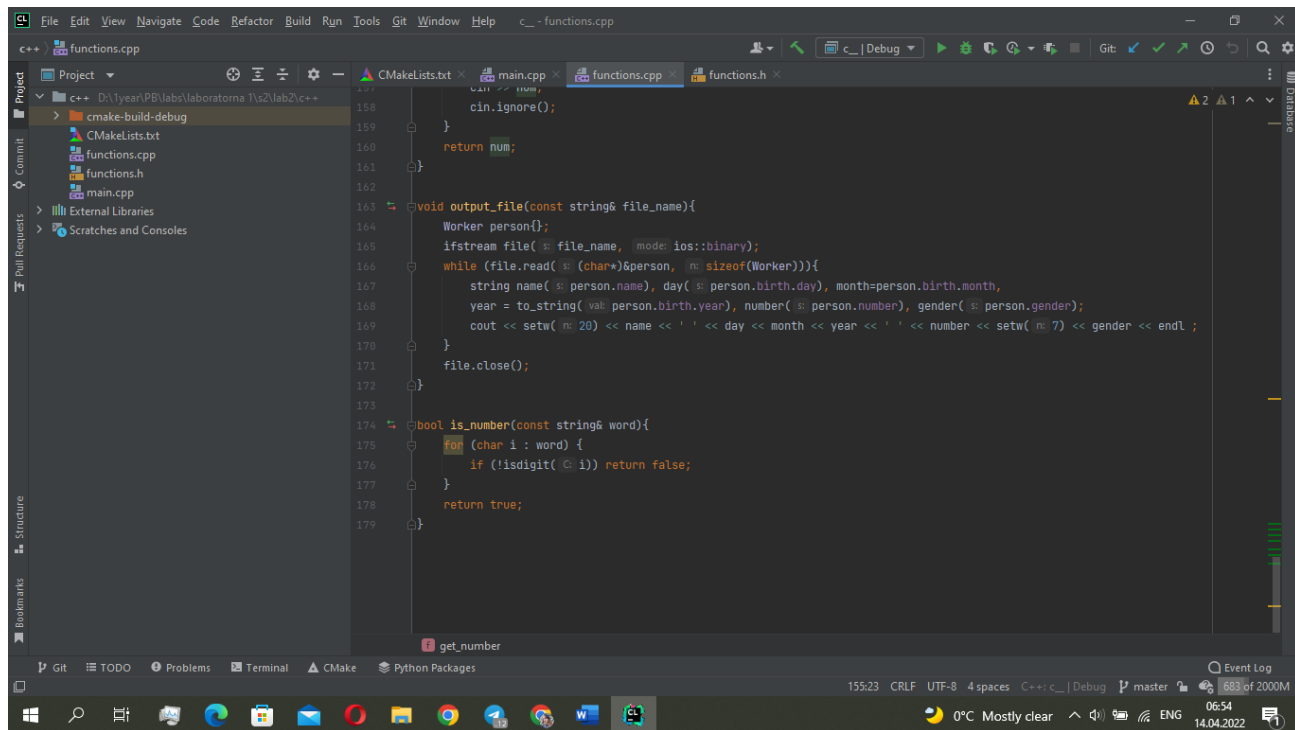
The screenshot shows the Visual Studio Code editor with the file `functions.cpp` open. The editor displays the implementation of two functions: `get_month()` and `get_year()`. The `get_month()` function prompts the user for a month number (1-12) and returns the month name as a string. The `get_year()` function prompts the user for a year and returns it as an integer. The project structure on the left shows a CMakeLists.txt file and a main.cpp file. The status bar at the bottom indicates the current file is `functions.cpp` and the editor is in Debug mode.

```
118 string get_month(){
119     string m;
120     cout << "Month of birth (number): ";
121     cin >> m;
122     cin.ignore();
123     while (!is_number(word: m) or stoi(str: m) < 1 or stoi(str: m) > 12){
124         cout << "Wrong input. Enter a number from 1 to 31: ";
125         cin >> m;
126         cin.ignore();
127     }
128     if (stoi(str: m) < 10) m = '0' + m;
129     return m+'.';
130 }
131
132 int get_year(){
133     string y;
134     cout << "Enter year of birth: ";
135     cin >> y;
136     cin.ignore();
137     while (!is_number(word: y) or stoi(str: y) < 1900 or stoi(str: y) > 2022) {
138         cout << "Wrong input. Try again: ";
139         cin >> y;
140         cin.ignore();
141     }
142     return stoi(str: y);
143 }
144
```

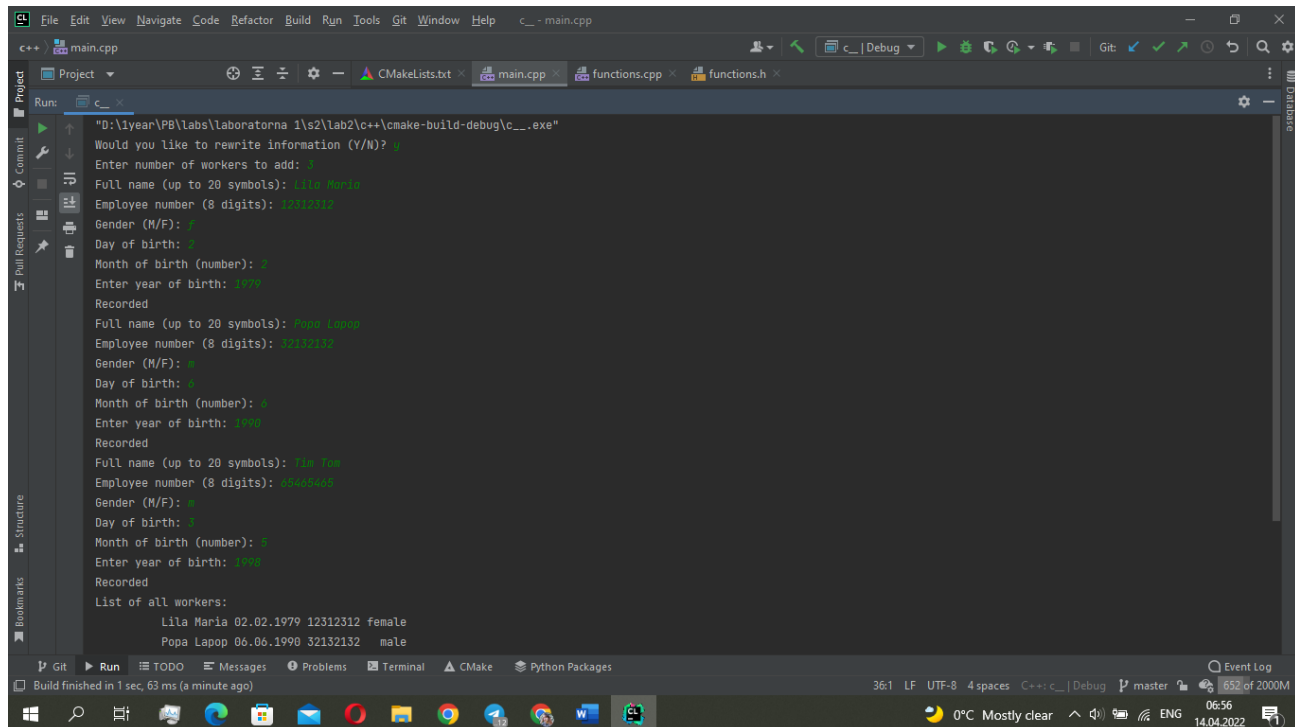
The screenshot shows the Visual Studio Code editor with the file `functions.cpp` open. The editor displays the implementation of two functions: `get_number()` and `output_file()`. The `get_number()` function prompts the user for an 8-digit employee number and returns it as a string. The `output_file()` function is a placeholder for writing to a file. The project structure on the left shows a CMakeLists.txt file and a main.cpp file. The status bar at the bottom indicates the current file is `functions.cpp` and the editor is in Debug mode.

```
141 }
142 return stoi(str: y);
143 }
144
145 string get_number(){
146     string num;
147     cout << "Employee number (8 digits): ";
148     cin >> num;
149     cin.ignore();
150     while (num.length() != 8){ //check if there are 8 chars
151         cout << "Wrong length. Enter 8 digits: ";
152         cin >> num;
153         cin.ignore();
154     }
155     while (!is_number(word: num)){ //check if all chars are digits
156         cout << "All chars have to be digits. Enter 8 digits: ";
157         cin >> num;
158         cin.ignore();
159     }
160     return num;
161 }
162
163 void output_file(const string& file_name){
164     Worker person{};
165 }

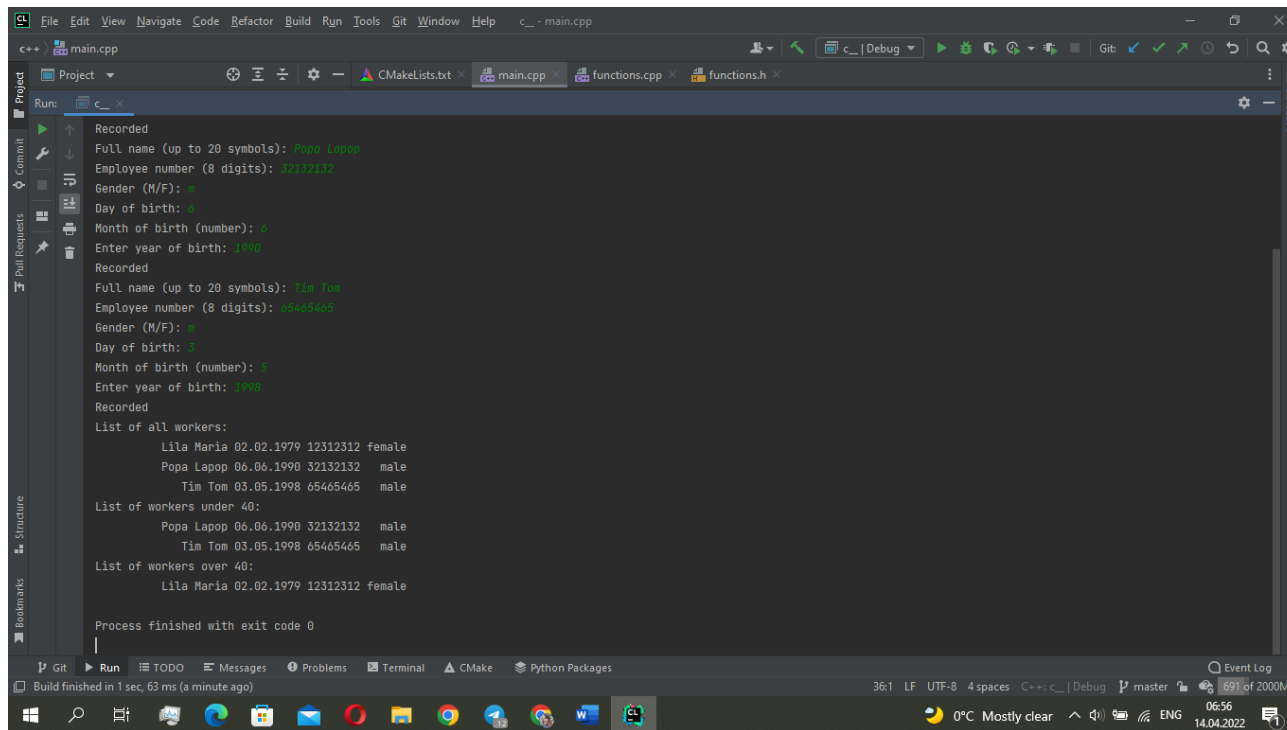
```



```
157     cin >> num;
158     cin.ignore();
159 }
160 return num;
161 }
162
163 void output_file(const string& file_name){
164     Worker person{};
165     ifstream file( file_name, mode::ios::binary);
166     while (file.read( (char*)&person, sizeof(Worker))){
167         string name( person.name), day( person.birth.day), month=person.birth.month,
168             year = to_string( person.birth.year), number( person.number), gender( person.gender);
169         cout << setw( 20) << name << ' ' << day << month << year << ' ' << number << setw( 7) << gender << endl ;
170     }
171     file.close();
172 }
173
174 bool is_number(const string& word){
175     for (char i : word) {
176         if (!isdigit( i)) return false;
177     }
178     return true;
179 }
```

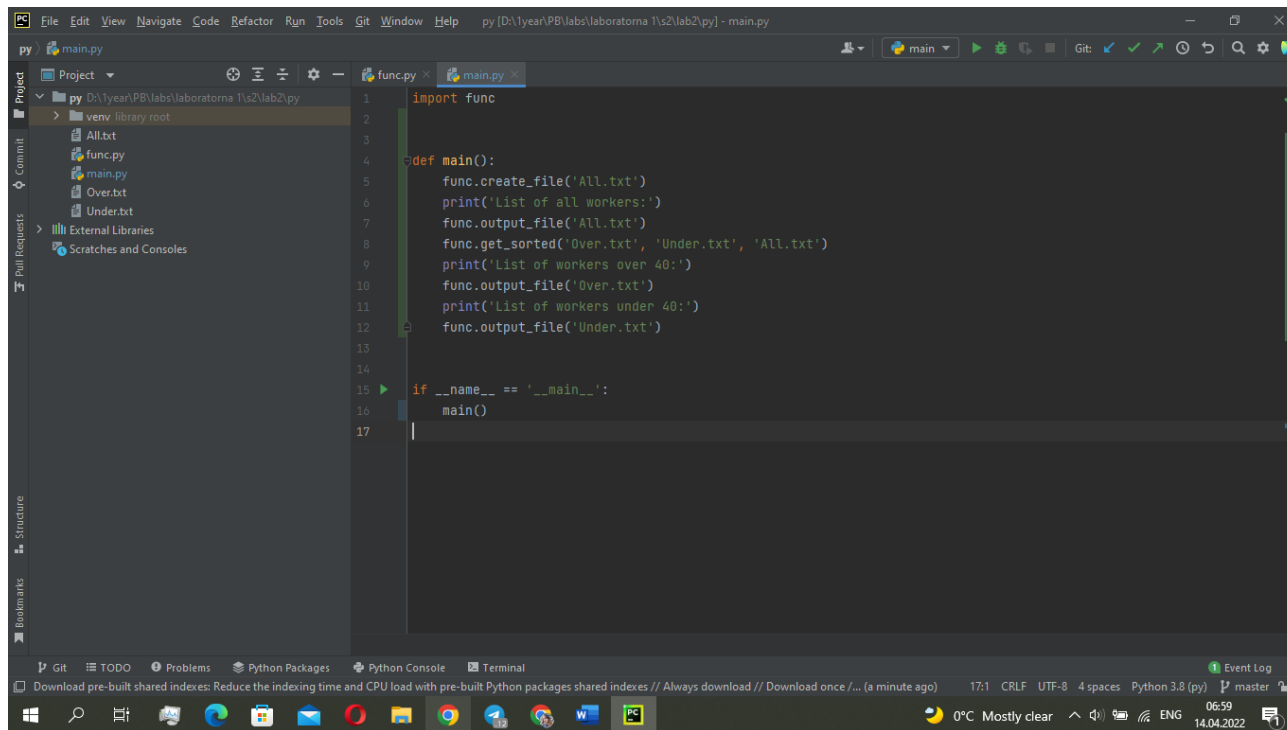


```
Run: c_
"D:\1year\PB\labs\laboratorna 1\s2\lab2\c++\cmake-build-debug\c_ .exe"
Would you like to rewrite information (Y/N)? y
Enter number of workers to add: 3
Full name (up to 20 symbols): Lila Maria
Employee number (8 digits): 12312312
Gender (M/F): f
Day of birth: 2
Month of birth (number): 2
Enter year of birth: 1979
Recorded
Full name (up to 20 symbols): Popa Lapop
Employee number (8 digits): 32132132
Gender (M/F): m
Day of birth: 6
Month of birth (number): 6
Enter year of birth: 1990
Recorded
Full name (up to 20 symbols): Popa Popa
Employee number (8 digits): 99999999
Gender (M/F): m
Day of birth: 1
Month of birth (number): 1
Enter year of birth: 1990
Recorded
List of all workers:
Lila Maria 02.02.1979 12312312 female
Popa Lapop 06.06.1990 32132132 male
```

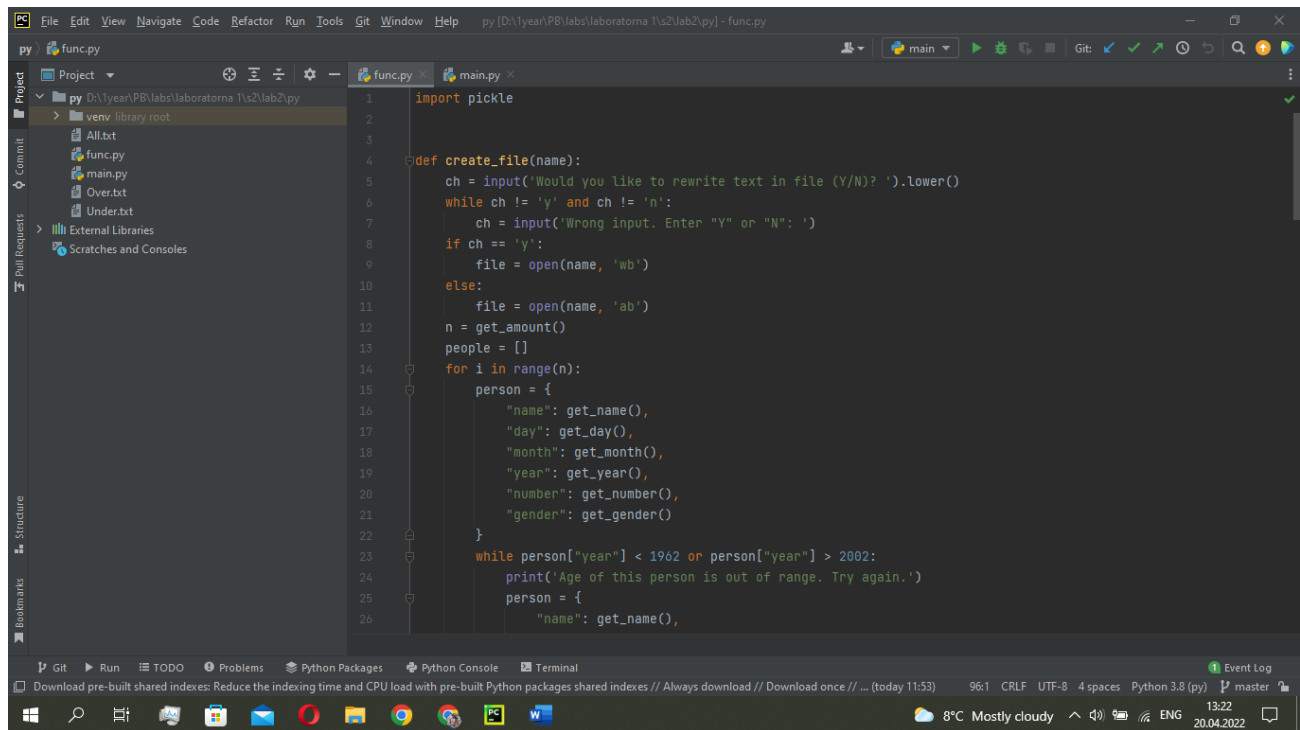


```
Recorded
Full name (up to 20 symbols): Popa Lapop
Employee number (8 digits): 32132132
Gender (M/F): f
Day of birth: 2
Month of birth (number): 2
Enter year of birth: 1979
Recorded
Full name (up to 20 symbols): Tim Tom
Employee number (8 digits): 65465465
Gender (M/F): m
Day of birth: 3
Month of birth (number): 5
Enter year of birth: 1998
Recorded
List of all workers:
      Lila Maria 02.02.1979 12312312 female
      Popa Lapop 06.06.1990 32132132 male
      Tim Tom 03.05.1998 65465465 male
List of workers under 40:
      Popa Lapop 06.06.1990 32132132 male
      Tim Tom 03.05.1998 65465465 male
List of workers over 40:
      Lila Maria 02.02.1979 12312312 female
Process finished with exit code 0
```

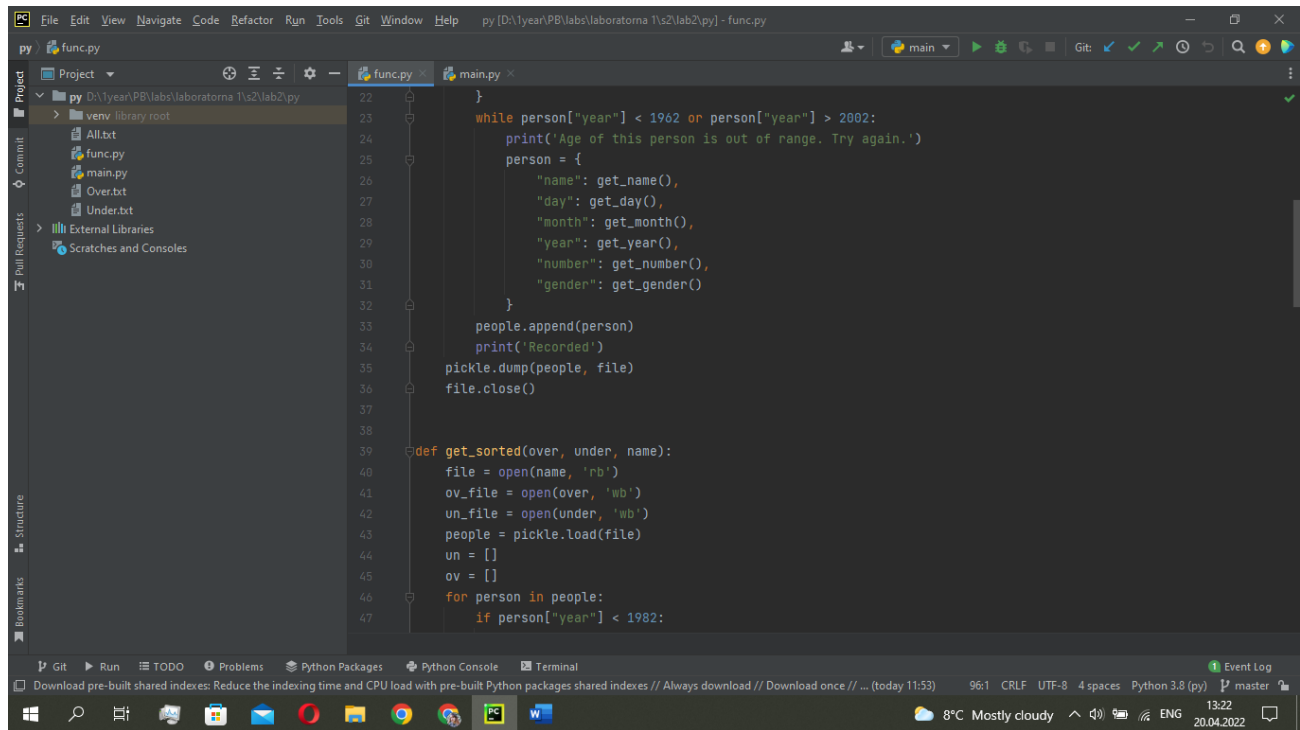
Виконання на Python



```
1 import func
2
3
4 def main():
5     func.create_file('All.txt')
6     print('List of all workers:')
7     func.output_file('All.txt')
8     func.get_sorted('Over.txt', 'Under.txt', 'All.txt')
9     print('List of workers over 40:')
10    func.output_file('Over.txt')
11    print('List of workers under 40:')
12    func.output_file('Under.txt')
13
14
15 if __name__ == '__main__':
16     main()
17
```

```
1 import pickle
2
3
4 def create_file(name):
5     ch = input('Would you like to rewrite text in file (Y/N)? ').lower()
6     while ch != 'y' and ch != 'n':
7         ch = input('Wrong input. Enter "Y" or "N": ')
8     if ch == 'y':
9         file = open(name, 'wb')
10    else:
11        file = open(name, 'ab')
12    n = get_amount()
13    people = []
14    for i in range(n):
15        person = {
16            "name": get_name(),
17            "day": get_day(),
18            "month": get_month(),
19            "year": get_year(),
20            "number": get_number(),
21            "gender": get_gender()
22        }
23        while person["year"] < 1962 or person["year"] > 2002:
24            print('Age of this person is out of range. Try again.')
25            person = {
26                "name": get_name(),
```



```
22    }
23    while person["year"] < 1962 or person["year"] > 2002:
24        print('Age of this person is out of range. Try again.')
25    person = {
26        "name": get_name(),
27        "day": get_day(),
28        "month": get_month(),
29        "year": get_year(),
30        "number": get_number(),
31        "gender": get_gender()
32    }
33    people.append(person)
34    print('Recorded')
35    pickle.dump(people, file)
36    file.close()
37
38
39 def get_sorted(over, under, name):
40     file = open(name, 'rb')
41     ov_file = open(over, 'wb')
42     un_file = open(under, 'wb')
43     people = pickle.load(file)
44     un = []
45     ov = []
46     for person in people:
47         if person["year"] < 1982:
```

```
py [D:\1year\PB\labs\laboratorna 1\s2\lab2\py] - func.py
Project
  py D:\1year\PB\labs\laboratorna 1\s2\lab2\py
    venv library root
    All.txt
    func.py
    main.py
    Over.txt
    Under.txt
  External Libraries
  Scratches and Consoles
  Git
  TODO
  Problems
  Python Packages
  Python Console
  Terminal
  Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once ... (a minute ago)
  96:31 CRLF UTF-8 4 spaces Python 3.8 (py) master
  0°C Mostly clear 07:00 14.04.2022
```

```
19 file.close()
20
21
22 def get_sorted(over, under, name):
23     file = open(name, 'rb')
24     ov_file = open(over, 'wb')
25     un_file = open(under, 'wb')
26     text = file.read().decode()
27     lines = text.splitlines()
28     for person in lines:
29         year = int(person[27:31])
30         if year < 1982:
31             ov_file.write(person.encode())
32         else:
33             un_file.write(person.encode())
34     file.close()
35     ov_file.close()
36     un_file.close()
37
38
39 def get_amount():
40     num = input('Enter number of workers to add: ')
41     while not num.isdigit():
42         num = input('Enter a positive integer: ')
43     return int(num)
```

```
py [D:\1year\PB\labs\laboratorna 1\s2\lab2\py] - func.py
Project
  py D:\1year\PB\labs\laboratorna 1\s2\lab2\py
    venv library root
    All.txt
    func.py
    main.py
    Over.txt
    Under.txt
  External Libraries
  Scratches and Consoles
  Git
  TODO
  Problems
  Python Packages
  Python Console
  Terminal
  Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once ... (2 minutes ago)
  96:31 CRLF UTF-8 4 spaces Python 3.8 (py) master
  0°C Mostly clear 07:00 14.04.2022
```

```
43     return int(num)
44
45
46 def get_name():
47     name = input('Full name (up to 20 symbols): ')
48     while len(name) > 20:
49         name = input('Too Long. Try again: ')
50     name = ' ' * (20 - len(name)) + name + ' '
51     return name
52
53
54 def get_number():
55     number = input('Employee number: ')
56     while not number.isdigit() or len(number) != 8:
57         number = input('Wrong input. Enter 8 digits: ')
58     return number + ' '
59
60
61 def get_gender():
62     gen = input('Gender (M/F): ').lower()
63     while gen != 'm' and gen != 'f':
64         gen = input('Wrong input. Enter letter M for male or F for female: ')
65     if gen == 'm':
66         return ' male\n'
67     else:
68         return 'female\n'
```

```
py [D:\1year\PB\labs\laboratorna 1\vs2\lab2\py] - func.py

Project
  py D:\1year\PB\labs\laboratorna 1\vs2\lab2\py
    venv library root
    All.txt
    func.py
    main.py
    Over.txt
    Under.txt
  External Libraries
  Scratches and Consoles

func.py
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92

main.py
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92

def get_day():
    day = input('Day of birth: ')
    while not day.isdigit() or int(day) > 31:
        day = input('Wrong input. Enter a number from 1 to 31: ')
    day = '0' * (2 - len(day)) + day + '.'
    return day

def get_month():
    month = input('Month of birth (number): ')
    while not month.isdigit() or int(month) > 12:
        month = input('Wrong input. Enter a number from 1 to 12: ')
    month = '0' * (2 - len(month)) + month + '.'
    return month

def get_year():
    year = input('Year of birth: ')
    while not year.isdigit() or int(year) < 1900 or int(year) > 2022:
        year = input('Wrong input. Try again: ')
    return year + ' '

output_file()
```

```
py [D:\1year\PB\labs\laboratorna 1\vs2\lab2\py] - func.py

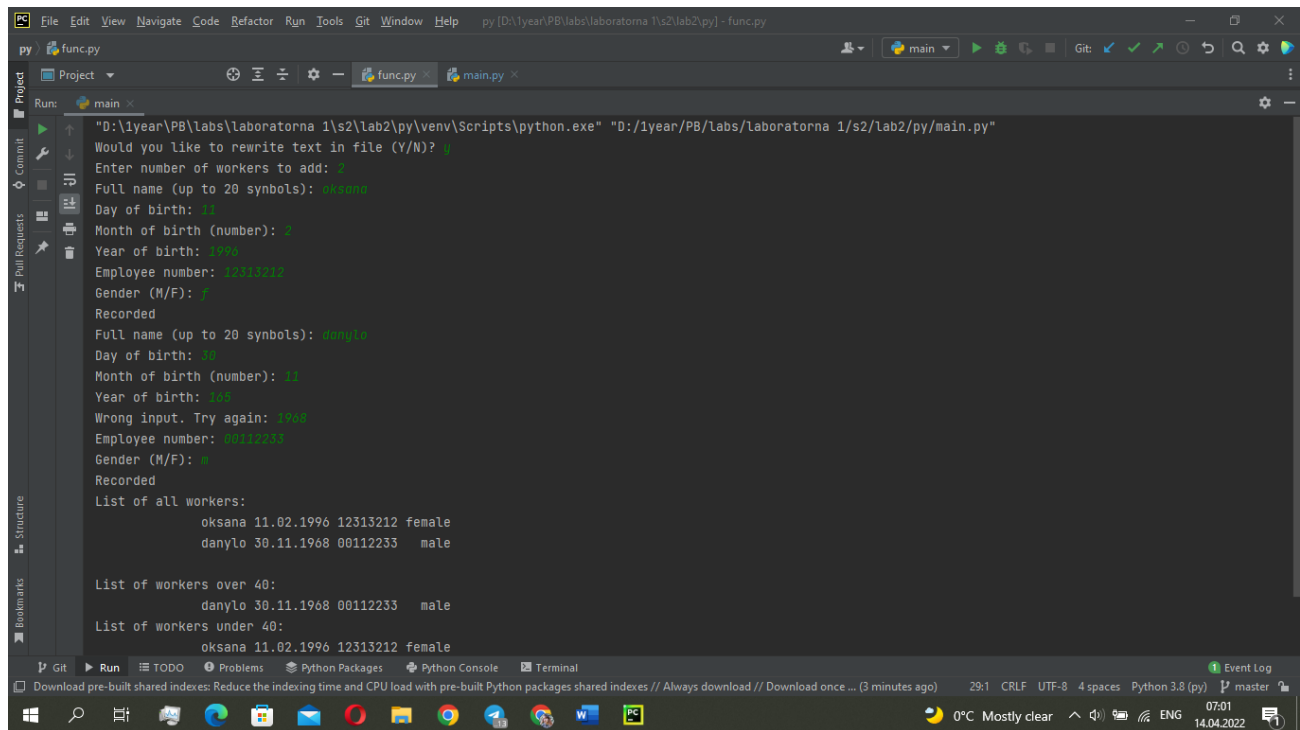
Project
  py D:\1year\PB\labs\laboratorna 1\vs2\lab2\py
    venv library root
    All.txt
    func.py
    main.py
    Over.txt
    Under.txt
  External Libraries
  Scratches and Consoles

main.py
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120

def get_month():
    month = input('Month of birth (number): ')
    while not month.isdigit() or int(month) > 12:
        month = input('Wrong input. Enter a number from 1 to 12: ')
    month = '0' * (2 - len(month)) + month
    return month

def get_year():
    year = input('Year of birth: ')
    while not year.isdigit() or int(year) < 1900 or int(year) > 2022:
        year = input('Wrong input. Try again: ')
    return int(year)

def output_file(name):
    file = open(name, 'rb')
    people = pickle.load(file)
    for person in people:
        print(person["name"] + ' ' + person["day"] + '.' + person["month"] + '.' + str(person["year"]) + ' ' +
              person["number"] + ' ' + person["gender"])
    file.close()
```



```
py [D:\1year\PB\labs\laboratorna 1\s2\lab2\py] - func.py
Run: main
"D:\1year\PB\labs\laboratorna 1\s2\lab2\py\venv\Scripts\python.exe" "D:/1year/PB/labs/laboratorna 1/s2/lab2/py/main.py"
Would you like to rewrite text in file (Y/N)? y
Enter number of workers to add: 2
Full name (up to 20 symbols): oksana
Day of birth: 11
Month of birth (number): 02
Year of birth: 1996
Employee number: 12313212
Gender (M/F): f
Recorded
Full name (up to 20 symbols): danylo
Day of birth: 30
Month of birth (number): 11
Year of birth: 1968
Wrong input. Try again: 1968
Employee number: 00112233
Gender (M/F): m
Recorded
List of all workers:
    oksana 11.02.1996 12313212 female
    danylo 30.11.1968 00112233 male

List of workers over 40:
    danylo 30.11.1968 00112233 male

List of workers under 40:
    oksana 11.02.1996 12313212 female
```

Висновки

Протягом лабораторної роботи було розглянуто роботу з бінарними файлами та використано отримані навички під час написання програм. В результаті роботи було створену програму, яка створює новий файл, що містить інформацію про співробітників, перевіряє на коректність введені дані, записує дані у файл, якщо вік співробітника від 20 до 60 років, а також створює 2 нових файли – зв співробітниками старше та молодше 40 років.