

ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2  
з дисципліни "Програмування та алгоритмічні мови "  
студентки групи ПС-24-1  
Гарт Анни Вікторівни  
кафедри ОМ та МК, ДНУ  
2024/2025

**Тема: «Базові операції вводу/виводу»**

## 1. Постановка задачі

Скласти звіт по виконаним завданням, та завантажити звіт та коди програм на GitHub.  
Для цього, створити репозиторій: TheBasicsOfProgrammingCppCourse

Та потім створити папку TheBasicsOfProgrammingCppCourse/Lab2/...

### 1) Задача Hello World

=====

Складіть програму "Hello World", котра  
виводить на екран текст привітання:  
"Hello, world!"

### 2) Задача SQUARE

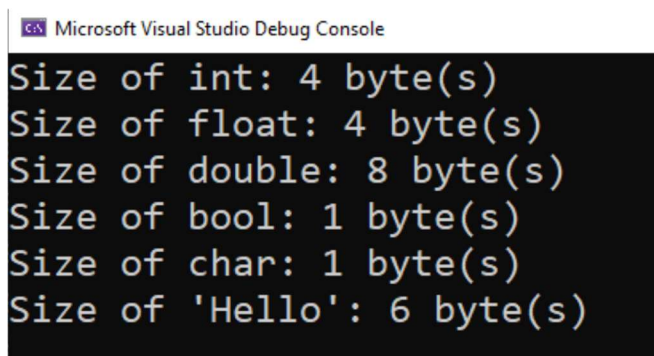
=====

Складіть програму SQUARE, котра виводить  
на екран квадрат в текстовому режимі (консоль).  
Для малювання квадрату використовуйте символи тексту.

### 3) Задача Bytes Calculator

=====

Bytes Calculator – що, за допомогою функції sizeof() виведе розміри базових типів  
на екран.



```
Size of int: 4 byte(s)
Size of float: 4 byte(s)
Size of double: 8 byte(s)
Size of bool: 1 byte(s)
Size of char: 1 byte(s)
Size of 'Hello': 6 byte(s)
```

#### 4) Задача ANKETA

=====

Складіть програму ANKETA, котра виводить на екран анкетні дані автора програми (прізвище, ім'я, по батькові, стать, дата народження, адреса, номер телефону, місце навчання, група, хоббі).

Приклад роботи програми:

Прізвище : Gates

Ім'я : William

По батькові : Henry

Стать : чоловіча

Дата народження : 1955-10-28

Адреса : One Microsoft Way, 1, NY, USA

Номер телефону : +380 12 3456789

Місце навчання : ФПМ, ДНУ ім. Олеся Гончара

Група : ПА-22-7

Хоббі : програмування

## 2. Вихідний текст програми розв'язку задачі Hello Word:

### Варіант 1

Task1\_1.cpp

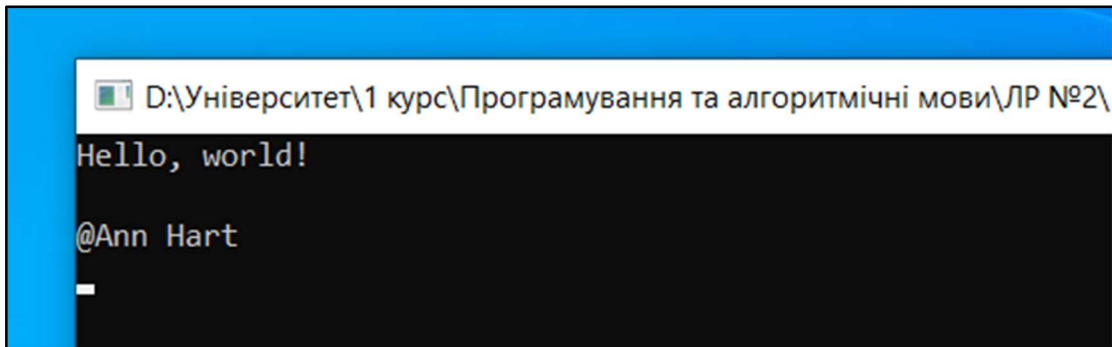
```
1 #include<iostream>
2 using namespace std;
3
4 int main(){
5     cout << "Hello, world!" << endl;
6     cout << "\n@Ann Hart" << endl;
7
8     cin.ignore();
9     cin.get();
10    return 0;
11 }
```

### Опис інтерфейсу програми.

1. `#include<iostream>` підключає бібліотеку для роботи з введенням та виведенням даних у мові C++
2. `using namespace std` дозволяє використовувати всі елементи простору (наприклад, `cout`, `endl`) без префіксу `std::`:
3. `int main(){...}` – це початкова точка виконання програми на C/C++; вона визначає головну функцію, яка повертає ціле число (`int`) після завершення роботи
4. `cout <<` (або `std::cout <<`) виводить на екран вказані дані; якщо потрібно вивести в одному рядку і текст, і значення змінних, то між ними теж потрібно ставити `<<`

5. `<< endl` – завершення рядка
6. `\n` – «абзац», ставиться у лапках серед тексту
6. текст, що записано у лапках `"Hello, world!"` і `"\n@Ann Hart"`, має виводитись без лапок і без `\n`
7. `cin.ignore()` – пропускає символ або кілька символів у буфері вводу
8. `cin.get()` – чекає на введення «Enter»; ця функція використовується мною, аби консоль не закривалась одразу після виконання програми
9. `return 0` вказує на те, що програма завершилась успішно

### Результат роботи програми:



### Варіант 2

```
Task1_2.cpp

1  #include<iostream>
2  using namespace std;
3
4  int main()
5  {
6      int choice;
7      cout << "Select language:" << endl;
8      cout << "1 - English" << endl;
9      cout << "2 - Deutsch" << endl;
10     cout << "3 - Ukrainian (transcription)" << endl;
11     cout << "Enter your choice (only number): ";
12     cin >> choice ;
13
14     switch (choice) {
15         case 1:
16             cout << "\n Hello, world!" << endl;
17             break;
18         case 2:
19             cout << "\n Hallo, Welt!" << endl;
20             break;
21         case 3:
22             cout << "\n Pryvit, cvit!" << endl;
23             break;
```

```

24         default:
25             cout << "\n ! Wrong choice !" << endl;
26             break;
27     }
28
29     cout << "\n@Ann Hart" << endl;
30
31     cin.ignore();
32     cin.get();
33     return 0;
34 }

```

### Опис інтерфейсу програми.

1. `int choice` додавання числової цілої змінної «choice»
2. `cin >> choice` присвоювання зазначеній змінній значення, введене у консолі; якщо значення не числове, програма працювати не буде
3. `switch (choice) {...}` перевіряє значення змінної «choice» та виконує відповідний блок коду для того `case`, який збігається з її значенням, або `default`

### Результати роботи програми:

```

D:\Університет\1 курс\Програмування та
Select language:
1 - English
2 - Deutsch
3 - Ukrainian (transcription)
Enter your choice (only number): 1

Hello, world!

@Ann Hart
_

```

```

D:\Університет\1 курс\Програмування та
Select language:
1 - English
2 - Deutsch
3 - Ukrainian (transcription)
Enter your choice (only number): 2

Hallo, Welt!

@Ann Hart
_

```

```

D:\Університет\1 курс\Програмування та
Select language:
1 - English
2 - Deutsch
3 - Ukrainian (transcription)
Enter your choice (only number): 3

Pryvit, cvit!

@Ann Hart
_

```

```

D:\Університет\1 курс\Програмування та
Select language:
1 - English
2 - Deutsch
3 - Ukrainian (transcription)
Enter your choice (only number): 23

! Wrong choice !

@Ann Hart
_

```

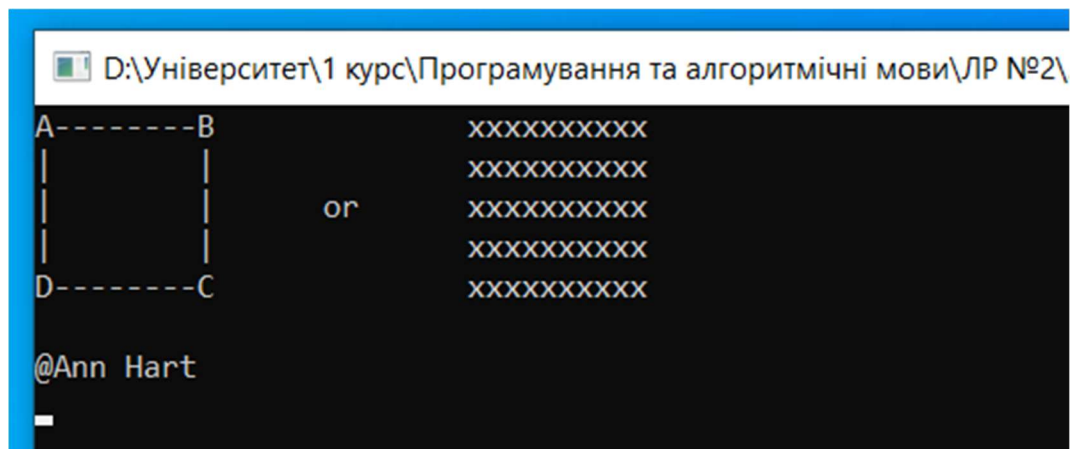
### 3. Вихідний текст програми розв'язку задачі SQUARE:

#### Варіант 1

```
Task2_1.cpp

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      cout << "A-----B" << "xxxxxxxxxxx" << endl;
6      cout << "|          |" << "xxxxxxxxxxx" << endl;
7      cout << "|          |" << "or" << "xxxxxxxxxxx" << endl;
8      cout << "|          |" << "xxxxxxxxxxx" << endl;
9      cout << "D-----C" << "xxxxxxxxxxx" << endl;
10
11     cout << "\n@Ann Hart" << endl;
12
13     cin.ignore();
14     cin.get();
15     return 0;
16 }
```

#### Результат роботи програми:



```
D:\Університет\1 курс\Програмування та алгоритмічні мови\ЛР №2\
A-----B          xxxxxxxxxxxx
|          |      xxxxxxxxxxxx
|          |      or  xxxxxxxxxxxx
|          |      xxxxxxxxxxxx
D-----C          xxxxxxxxxxxx

@Ann Hart
_
```

## Вариант 2

### Task2\_2.cpp

```
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  int main() {
6      int n;
7
8      cout << "Enter the length of the side of the square (integer): ";
9      cin >> n;
10
11     if (n > 0) {
12         /* Side AB */
13         /* Sq_1 */
14         cout << "\nA";
15         for (int a = 0; a < n; a++) {
16             cout << "---";
17         }
18         cout << "B          ";
19
20         /* Sq_2 */
21         cout << "x";
22         for (int a = 0; a < n; a++) {
23             cout << "xxx";
24         }
25         cout << "x" << endl;
26
27         /* Side AD and BC */
28         if (n <= 2) {
29             for (int b = 0; b < n; b++) {
30                 /* Sq_1 */
31                 cout << "|";
32                 for (int d = 0; d < n; d++) {
33                     cout << "   ";
34                 }
35                 cout << "|          ";
36
37                 /* Sq_2 */
38                 cout << "x";
39                 for (int d = 0; d < n; d++) {
40                     cout << "xxx";
41                 }
42                 cout << "x" << endl;
43             }
44         }
45         else {
46             int up_lim = 6;
47             if (n <= up_lim) {
```

```

48         for (int b = -1; b < n; b++) {
49             /* Sq_1 */
50             cout << "|";
51             for (int d = 0; d < n; d++) {
52                 cout << " ";
53             }
54             cout << "|";
55
56             /* Sq_2 */
57             cout << "x";
58             for (int d = 0; d < n; d++) {
59                 cout << "xxx";
60             }
61             cout << "x" << endl;
62         }
63     }
64     else {
65         int b = -1;
66         while (n > up_lim) {
67             up_lim += 4;
68             b -= 1;
69             if (n <= up_lim) {
70                 for (; b < n; b++) {
71                     /*Sq_1*/
72                     cout << "|";
73                     for (int d = 0; d < n; d++) {
74                         cout << " ";
75                     }
76                     cout << "|";
77
78                     /*Sq_2*/
79                     cout << "x";
80                     for (int d = 0; d < n; d++) {
81                         cout << "xxx";
82                     }
83                     cout << "x" << endl;
84                 }
85             }
86         }
87     }
88 }
89
90 /* Side CD */
91 /* Sq_1 */
92 cout << "D";
93 for (int c = 0; c < n; c++) {
94     cout << "---";
95 }
96 cout << "C";
97
98 /* Sq_2 */

```

```

99         cout << "x";
100         for (int c = 0; c < n; c++) {
101             cout << "xxx";
102         }
103         cout << "x" << endl;
104
105         cout << "\nP = 4 * AB = 4 * " << n << " = " << n * 4 << endl;
106         cout << "S = AB^2 = " << n << "^2 = " << n * n << endl;
107         cout << "AC = BD = sqrt(AB^2 + BC^2) = sqrt(2 * AB^2) = sqrt(2
            * " << n << "^2) = " << sqrt(2 * n * n) << endl;
108     }
109     else if (n == 0) {
110         cout << " ! Length can't be equal to zero !" << endl;
111     }
112     else {
113         cout << " ! Length can't be less than zero !" << endl;
114     }
115
116     cout << "\n@Ann Hart" << endl;
117
118     cin.ignore();
119     cin.get();
120     return 0;
121 }

```

## Опис інтерфейсу програми.

1. Щоб “намалювати” квадрат, потрібно переконатись, чи належить множині натуральних чисел значення змінної `n`, введеної користувачем.

Для цього введемо оператори: `if (n > 0) {...} ;`

`else if (n == 0) {...} ;`

`else {...} .`

2. Якщо перші дві умови не виконуються, то в консолі має бути виведено:

`! Length can't be less than zero !`.

3. Якщо  $n = 0$ , то в консолі має бути виведено:

`! Length can't be equal to zero !`.

4. Якщо  $n > 0$ , то:

а) будуємо сторону АВ:

- для квадрата №1:

- починаємо рядок і друкуємо вершину «А»;

- задаємо нову змінну `int a = 0`, у такому циклі `for (...) {...}`, що поки `a` буде менше значення змінної `n`, буде друкуватись «---», а змінна `a` з кожним разом буде збільшуватись на 1;



- друкуємо вершину «В» і рядок ми не закриваємо, оскільки квадрат №2 будемо будувати поруч, тому після «В» натискаємо ще 4 рази на «Tab».

- для квадрата №2:

- цей квадрат буде тільки з «х», тому замість «А» друкуємо «х»;
- задаємо знову змінну `int a = 0`, у такому циклі `for (...) {...}`, що поки `a` менше значення змінної `n`, буде друкуватись «xxx», а змінна `a` з кожним разом буде збільшуватись на 1;
- замість «В» друкуємо вершину «х» і завершуємо рядок.

б) будуємо сторони BC і AD:

- підбираючи різні значення `n`, було виявлено, що для побудови саме квадратів, а не прямокутників, у довжини має бути верхня межа, в залежності від значення змінної `n`. Тобто методом підбору, було визначено, що існують 3 варіанти верхніх меж довжини:

- коли  $n < 2$ :  $BC = AD = 2 * \llcorner \gg$  ;
- коли  $n \leq 6$ :  $BC = AD = \llcorner \gg * (n + 1)$ ;
- в інакшому випадку, верхня межа збільшується на 4 стільки  $j$ -разів, поки `n` не припинить перевищувати нову верхню межу:

$$BC = AD = \llcorner \gg * (n + 1 + j).$$

Наприклад,  $n = 11$ , тоді  $up\_lim = 6 + 4 \cdot 2 = 14$ , звідси  $j = 2$ , тоді

$$BC = AD = \llcorner \gg * (11 + 1 + 2) = \llcorner \gg \cdot 14,$$

тобто таких символів має бути 14 у вертикальній осі (перевірено ↓).

Розмірковуючи таким чином, було складено такий алгоритм:

```
if (n <= 2) {...}
else {
    int up_lim = 6;
    if (n <= 2) {...}
    else {
        int b = -1;
        while (n > up_lim) {
            up_lim += 4;
            b -= 1;
            if (n <= up_lim) {...}
        }
    }
}
```

Замість {...} будуються вертикальні сторони двох квадратів за алгоритмами, наведеними у пунктах нижче.

- задаємо нову змінну `int b`, у такому циклі `for (...) {...}`, що поки `b` буде менше значення змінної `n`:

(для квадрата №1)

- друкувалась бокова грань «|»;
- виконувався такий цикл `for (...) {...}`, що поки нова змінна `d` менше значення змінної `n`, буде друкуватись « » (3 пробіли), а змінна `d` з кожним разом буде збільшуватись на 1;
- знову друкувалась бокова грань «|» і чотири «Tab».

(для квадрата №2)

- друкувався «x»;
- виконувався такий цикл `for (...) {...}`, що поки нова змінна `d` менше значення змінної `n`, буде друкуватись «xxx», а змінна `d` з кожним разом буде збільшуватись на 1;
- знову друкувався «x» і рядок на цьому завершувався.

(для циклу)

- змінна `b` з кожним разом збільшувалась на 1;

в) будуємо сторону CD (аналогічно до сторони AB):

- для квадрата №1:
  - друкуємо вершину «D»;
  - задаємо нову змінну `int c = 0`, у такому циклі `for (...) {...}`, що поки `c` менше значення змінної `n`, буде друкуватись «---», а змінна `c` з кожним разом буде збільшуватись на 1;
  - друкуємо вершину «C» і чотири «Tab».
- для квадрата №2:
  - друкуємо «x»;
  - задаємо нову змінну `int c = 0`, у такому циклі `for (...) {...}`, що поки `c` менше значення змінної `n`, буде друкуватись «xxx», а змінна `c` з кожним разом буде збільшуватись на 1;
  - друкуємо «x» і завершуємо рядок.

г) на доданок, мною було написано блок коду, що рахує і виводить значення периметру, площі і діагоналі квадрату. Для обчислення діагоналі, що вимагало вилучення кореня, довелося на початку програми підключити бібліотеку для роботи з математичними операціями (знахідка з ін-ету).

## Результати роботи програми:

1) приклад з від'ємним числом:

```
D:\Університет\1 курс\Програмування та алгоритмічні мови\ЛР №2\2. Square\Version 2\main.exe
Enter the length of the side of the square (integer): -7
! Length can't be less than zero !

@Ann Hart
_
```

2) приклад з нулем:

```
D:\Університет\1 курс\Програмування та алгоритмічні мови\ЛР №2\2. Square\Version 2\main.exe
Enter the length of the side of the square (integer): 0
! Length can't be equal to zero !

@Ann Hart
_
```

3) приклад з  $n = 11$ :

```
D:\Університет\1 курс\Програмування та алгоритмічні мови\ЛР №2\2. Square\Version 2\main.exe
Enter the length of the side of the square (integer): 11

A-----B
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
|                                     |
D-----C

P = 4 * AB = 4 * 11 = 44
S = AB^2 = 11^2 = 121
AC = BD = sqrt(AB^2 + BC^2) = sqrt(2 * AB^2) = sqrt(2 * 11^2) = 15.5563

@Ann Hart
_
```

#### 4. Вихідний текст програми розв'язку задачі Bytes Calculator:

```
Task3.cpp

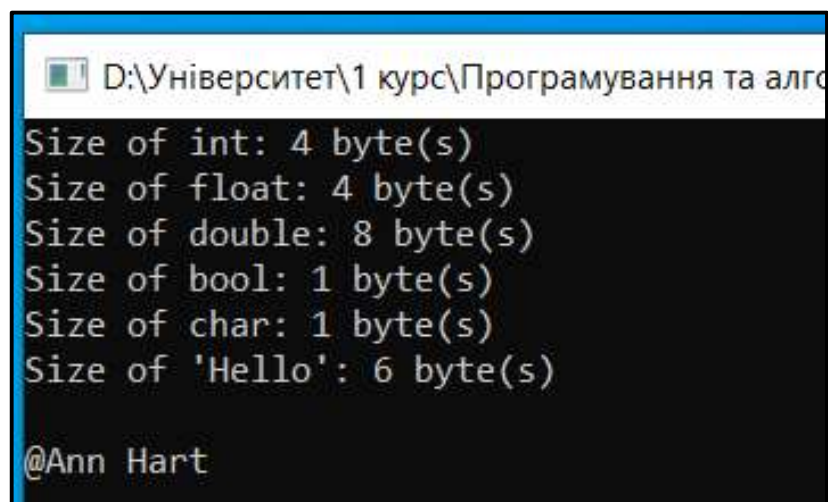
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5
6      cout << "Size of int: " << sizeof(int) << " byte(s)" << endl;
7      cout << "Size of float: " << sizeof(float) << " byte(s)" << endl;
8      cout << "Size of double: " << sizeof(double) << " byte(s)" << endl;
9      cout << "Size of bool: " << sizeof(bool) << " byte(s)" << endl;
10     cout << "Size of char: " << sizeof(char) << " byte(s)" << endl;
11     cout << "Size of 'Hello': " << sizeof("Hello") << " byte(s)" << endl;
12
13     cout << "\n@Ann Hart" << endl;
14
15     cin.ignore();
16     cin.get();
17     return 0;
18 }
```

##### Опис інтерфейсу програми.

sizeof(...) визначає кількість байтів, яку займає тип даних `int` / `float` / `double` / `bool` / `char` у пам'яті.

sizeof("Hello") визначає розмір рядка, включаючи нульовий термінуючий символ (\0), тому результат буде 6 байтів.

##### Результат роботи програми:



```
D:\Університет\1 курс\Програмування та алго
Size of int: 4 byte(s)
Size of float: 4 byte(s)
Size of double: 8 byte(s)
Size of bool: 1 byte(s)
Size of char: 1 byte(s)
Size of 'Hello': 6 byte(s)

@Ann Hart
```

## 5. Вихідний текст програми розв'язку задачі ANKETA:

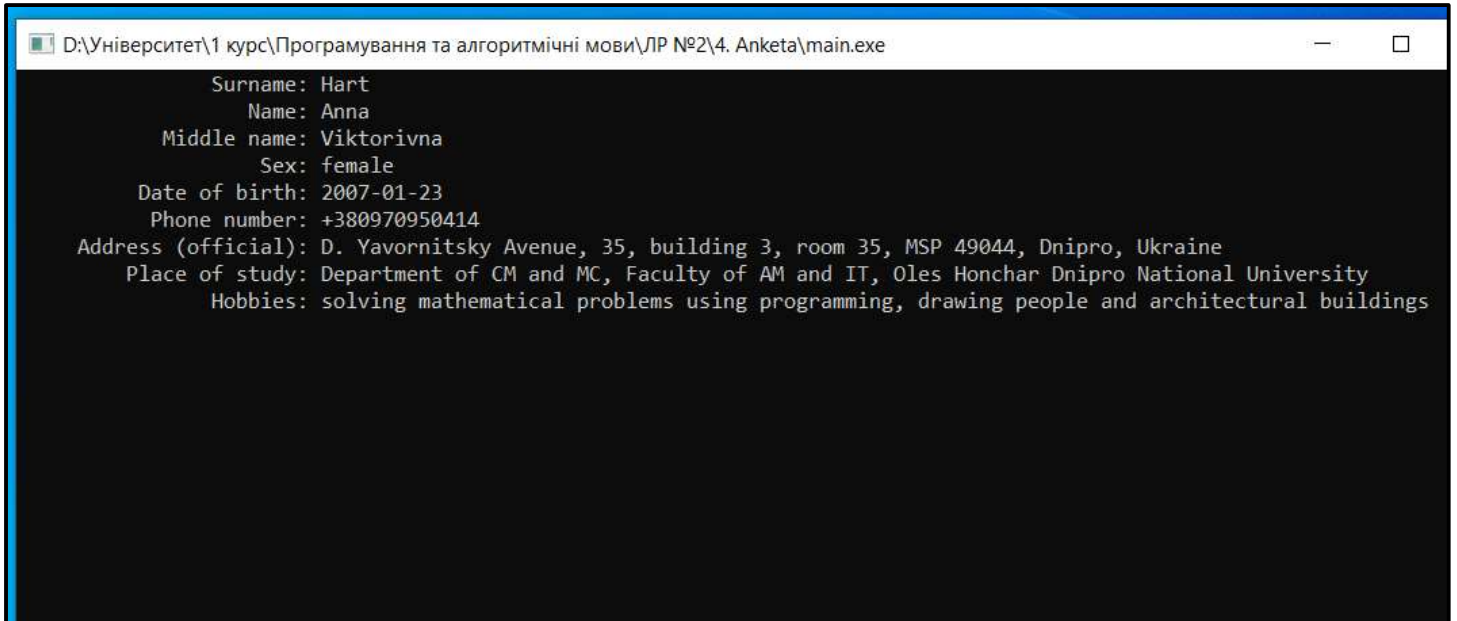
```
Task4.cpp
1  #include <iostream>
2  #include <iomanip> // для setw()
3  using namespace std;
4
5  int main(){
6      cout << setw(25) << right << "Surname: ";
7      cout << "Hart" << endl;
8
9      cout << setw(25) << right << "Name: ";
10     cout << "Anna" << endl;
11
12     cout << setw(25) << right << "Middle name: ";
13     cout << "Viktorivna" << endl;
14
15     cout << setw(25) << right << "Sex: ";
16     cout << "female" << endl;
17
18     cout << setw(25) << right << "Date of birth: ";
19     cout << "2007-01-23" << endl;
20
21     cout << setw(25) << right << "Phone number: ";
22     cout << "+380970950414" << endl;
23
24     cout << setw(25) << right << "Address (official): ";
25     cout << "D. Yavornitsky Avenue, 35, building 3, room 35, MSP
49044, Dnipro, Ukraine" << endl;
26
27     cout << setw(25) << right << "Place of study: ";
28     cout << "Department of CM and MC, Faculty of AM and IT, Oles
Honchar Dnipro National University" << endl;
29
30     cout << setw(25) << right << "Hobbies: ";
31     cout << "solving mathematical problems using programming, drawing
people and architectural buildings" << endl;
32
33     cin.ignore();
34     cin.get();
35     return 0;
36 }
```

## Опис інтерфейсу програми.

Вводимо бібліотеку `#include <iomanip>` для подальшого використання команди `setw(...)` (задавання ширини текстового поля). Маніпулятор `right` було використано для вирівнювання тексту по правому краю.

Рядок, що містить ці маніпулятори, не завершується для того, щоб текст, вказаний на наступному рядку, друкувався одразу після «питання».

## Результат роботи програми:



```
Surname: Hart
Name: Anna
Middle name: Viktorivna
Sex: female
Date of birth: 2007-01-23
Phone number: +380970950414
Address (official): D. Yavornitsky Avenue, 35, building 3, room 35, MSP 49044, Dnipro, Ukraine
Place of study: Department of CM and MC, Faculty of AM and IT, Oles Honchar Dnipro National University
Hobbies: solving mathematical problems using programming, drawing people and architectural buildings
```