

Звіт

З курсу «Навчальна практика: Мова програмування C++»

Студента 3 курсу, КІ-СА

Білецького Павла Володимировича

Розділ: Essentials

Lab 2.3.7 Do it yourself: Fibonacci sequence

Умова:

Написати програму яка виводить n-не число Фібоначчі. Перші 5 чисел :

1, 1, 2, 3, 5

Як видно перші 2 числа = 1, а кожне наступне = суммі 2 попередніх

Код програми:

```
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;
int main(void) {
    unsigned long long int n, a = 0, b = 0, c;
    cout << " Fibonacci sequence"<<endl;
    cout << "Enter number : ";
    cin >> n;
    a = 1;
    b = 1;
    for (int i = 2; i < n; i++) {
        c = a + b;
        b = a;
        a = c;
    }
    cout << endl << c << endl;
    return 0;
}
```

Компіляція в Linux

```
root@ip-172-31-27-42:~# vi test2.cpp
root@ip-172-31-27-42:~# g++ ./test2.cpp
root@ip-172-31-27-42:~# ls
a.out test1.cpp test2.cpp
```

Приклади вхідних даних

Example input

8

Example output

21

Example input

20

Example output

6765

Example input

55

Example output

139583862445

```
root@ip-172-31-27-42:~# ./a.out
Fibonacci sequence
Enter number : 8

21
```

```
root@ip-172-31-27-42:~# ./a.out
Fibonacci sequence
Enter number : 20

6765
```

```
root@ip-172-31-27-42:~# ./a.out
Fibonacci sequence
Enter number : 55

139583862445
root@ip-172-31-27-42:~# █
```

Власні тести

```
root@ip-172-31-27-42:~# ./a.out
Fibonacci sequence
Enter number : 4

3
root@ip-172-31-27-42:~# █
```

```
root@ip-172-31-27-42:~# ./a.out
Fibonacci sequence
Enter number : 95

13493690561280548289
root@ip-172-31-27-42:~# █
```

Розділ: Advanced

Lab 1.3.8.1 Iterators - over-collection

Умова:

Користувач вводить число – якщо число непарне, до нього додається одиниця
Програма повинна створити 2 масиви, один з яких містить всі додатні непарні а
інший – всі парні числа аж до даного користувачем та вивести суму всіх
комбінацій елементів першого масива з другим при переборі їх елементів
спочатку в зростаючому а потім в спадному порядку

```
#include <iostream>
#include <vector>
using namespace std;
int main()
{
    int n;
    cout << "Max Value? ";
    cin >> n;
    if (n % 2 != 0)
    {
        n++;
    }
    vector<int> v1(n / 2);
    vector<int> v2(n / 2);
    int k = 0;
    for (int i = 0; i < v1.size(); i++)
    {
        k++;
        v1[i] = k;
        k++;
        v2[i] = k;
    }
    for (int i = 0; i < v1.size(); i++)
    {
        for (int j = 0; j < v2.size(); j++)
        {
            cout << v1[i]+v2[j]<< " " ;
        }
        cout << endl;
    }
    for (int i = v1.size()-1; i >= 0; i--)
    {
        for (int j = v2.size()-1; j >= 0; j--)
        {
            cout << v1[i] + v2[j] << " ";
        }
        cout << endl;
    }
    //your code
    return 0;
}
```

Компіляція в Linux

```

root@ip-172-31-27-42:~# g++ ./test1.cpp
root@ip-172-31-27-42:~# ls
a.out  test1.cpp
root@ip-172-31-27-42:~#

```

Приклади вхідних даних

Example input

1

Example output

3
3

Example input

2

Example output

3
3

Example input

5

Example output

3 5 7
5 7 9
7 9 11
11 9 7
9 7 5
7 5 3

```

root@ip-172-31-27-42:~# g++ ./test1.cpp
root@ip-172-31-27-42:~# ./a.out
Max Value? 1
3
3
root@ip-172-31-27-42:~# ./a.out
Max Value? 2
3
3
root@ip-172-31-27-42:~#

```

```
root@ip-172-31-27-42:~# ./a.out
Max Value? 5
3 5 7
5 7 9
7 9 11
11 9 7
9 7 5
7 5 3
root@ip-172-31-27-42:~#
```

Власні приклади

```
root@ip-172-31-27-42:~# ./a.out
Max Value? 10
3 5 7 9 11
5 7 9 11 13
7 9 11 13 15
9 11 13 15 17
11 13 15 17 19
19 17 15 13 11
17 15 13 11 9
15 13 11 9 7
13 11 9 7 5
11 9 7 5 3
```

```
root@ip-172-31-27-42:~# ./a.out
Max Value? 15
3 5 7 9 11 13 15 17
5 7 9 11 13 15 17 19
7 9 11 13 15 17 19 21
9 11 13 15 17 19 21 23
11 13 15 17 19 21 23 25
13 15 17 19 21 23 25 27
15 17 19 21 23 25 27 29
17 19 21 23 25 27 29 31
31 29 27 25 23 21 19 17
29 27 25 23 21 19 17 15
27 25 23 21 19 17 15 13
25 23 21 19 17 15 13 11
23 21 19 17 15 13 11 9
21 19 17 15 13 11 9 7
19 17 15 13 11 9 7 5
17 15 13 11 9 7 5 3
```