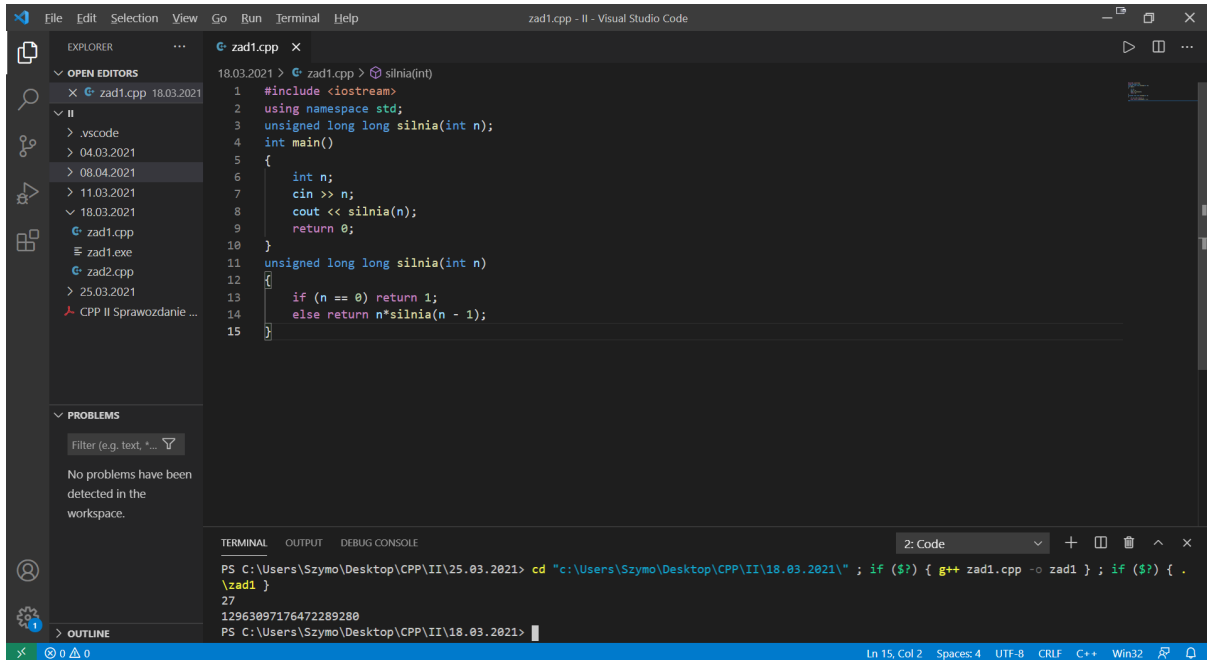


Szymon Dopieralski 6548

UWAGA: Niestety nie mogłem zbadać czasu wykonywania funkcji ani za pomocą biblioteki <chrono> ani <ctime>, ponieważ funkcje wykonywały się na tyle szybko, że zawsze pokazywało czas 0.

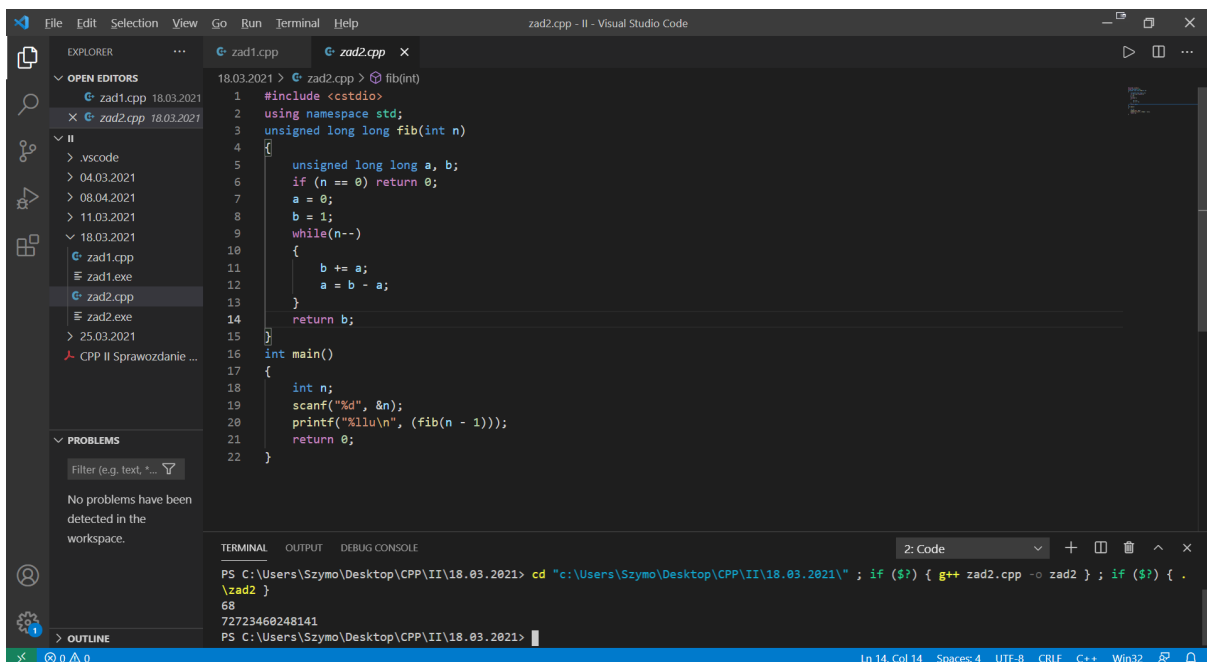
1. Napisz program, który wyliczy silnię 6, 16, 27, 68, 69, 125, 1000, 10000, 100000. Porównaj wyniki ze sobą



The screenshot shows the Visual Studio Code editor with a C++ file named `zad1.cpp`. The code implements a recursive function `silnia(int n)` to calculate factorials. The `main` function prompts the user to enter a number `n` and then calls `silnia(n)` to compute the result. The terminal shows the command to compile and run the program, and the output displays the factorial of 27, which is 12963097176472289280.

```
18.03.2021 > cd "c:\Users\Szymo\Desktop\CPP\II\18.03.2021\" ; if ($?) { g++ zad1.cpp -o zad1 } ; if ($?) { .\zad1 }
27
12963097176472289280
PS C:\Users\Szymo\Desktop\CPP\II\18.03.2021>
```

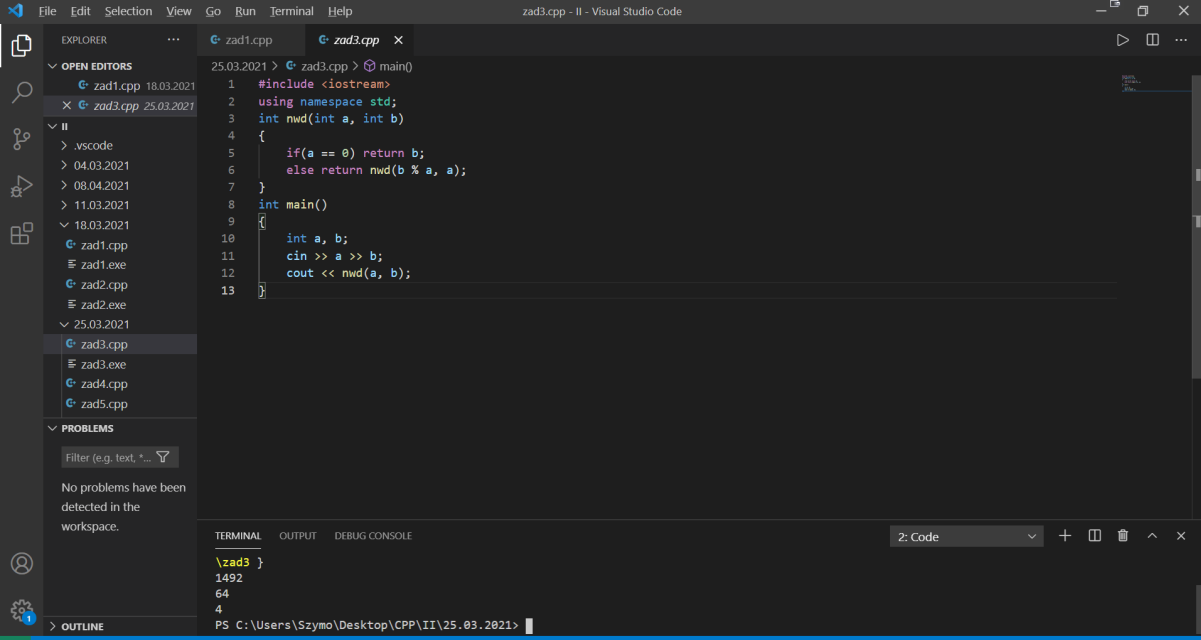
2. Napisz program, który wyliczy 6, 16, 27, 68, 69, 125, 1000, 10000, 100000 element ciągu Fibonacciego. Porównaj wyniki ze sobą.



The screenshot shows the Visual Studio Code editor with a C++ file named `zad2.cpp`. The code implements an iterative function `fib(int n)` to calculate the `n`-th element of the Fibonacci sequence. The `main` function prompts the user to enter a number `n` and then calls `fib(n)` to compute the result. The terminal shows the command to compile and run the program, and the output displays the 27th element of the Fibonacci sequence, which is 72723460248141.

```
18.03.2021 > cd "c:\Users\Szymo\Desktop\CPP\II\18.03.2021\" ; if ($?) { g++ zad2.cpp -o zad2 } ; if ($?) { .\zad2 }
68
72723460248141
PS C:\Users\Szymo\Desktop\CPP\II\18.03.2021>
```

3. Napisz program w którym zaimplementujesz rozwiązanie rekurencyjne algorytmu Euklidesa służącego do wyznaczania największego wspólnego dzielnika.



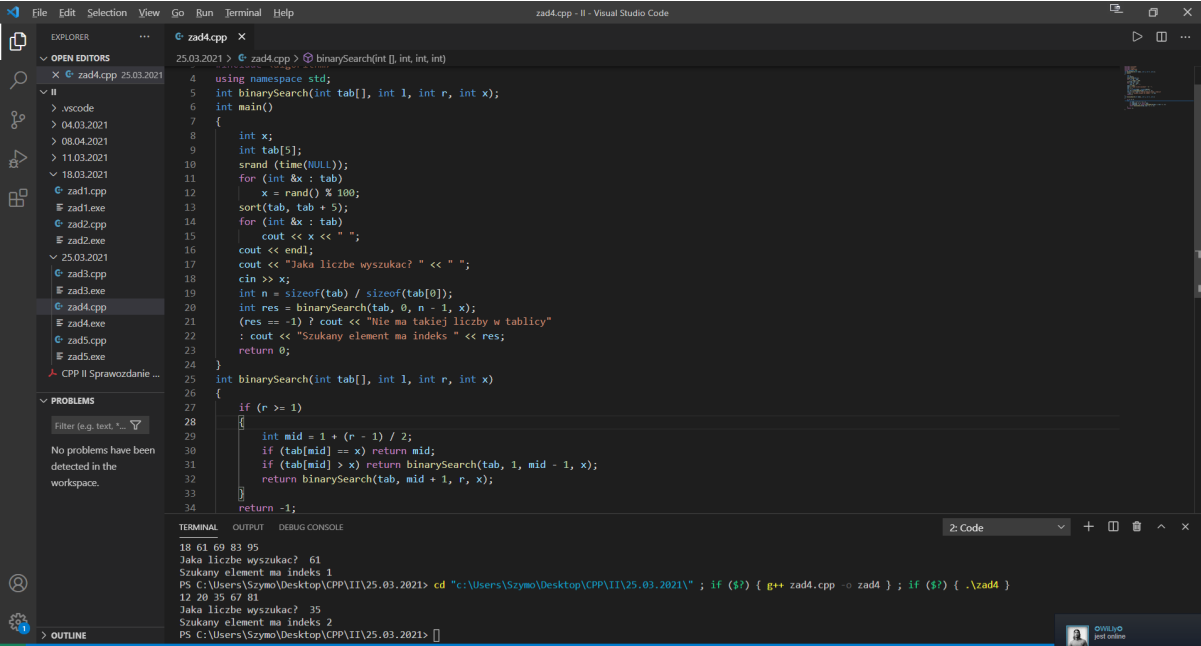
The screenshot shows the Visual Studio Code editor with the file `zad3.cpp` open. The code implements the recursive Euclidean algorithm to find the Greatest Common Divisor (GCD) of two numbers `a` and `b`. The `main` function takes input from the user and prints the result.

```
1 #include <iostream>
2 using namespace std;
3 int nwd(int a, int b)
4 {
5     if(a == 0) return b;
6     else return nwd(b % a, a);
7 }
8 int main()
9 {
10     int a, b;
11     cin >> a >> b;
12     cout << nwd(a, b);
13 }
```

The terminal output shows the execution of the program:

```
\zad3 }
1492
64
4
PS C:\Users\Szymo\Desktop\CPP\II\25.03.2021>
```

4. Napisz program, który zaimplementuje algorytm wyszukiwania binarnego.



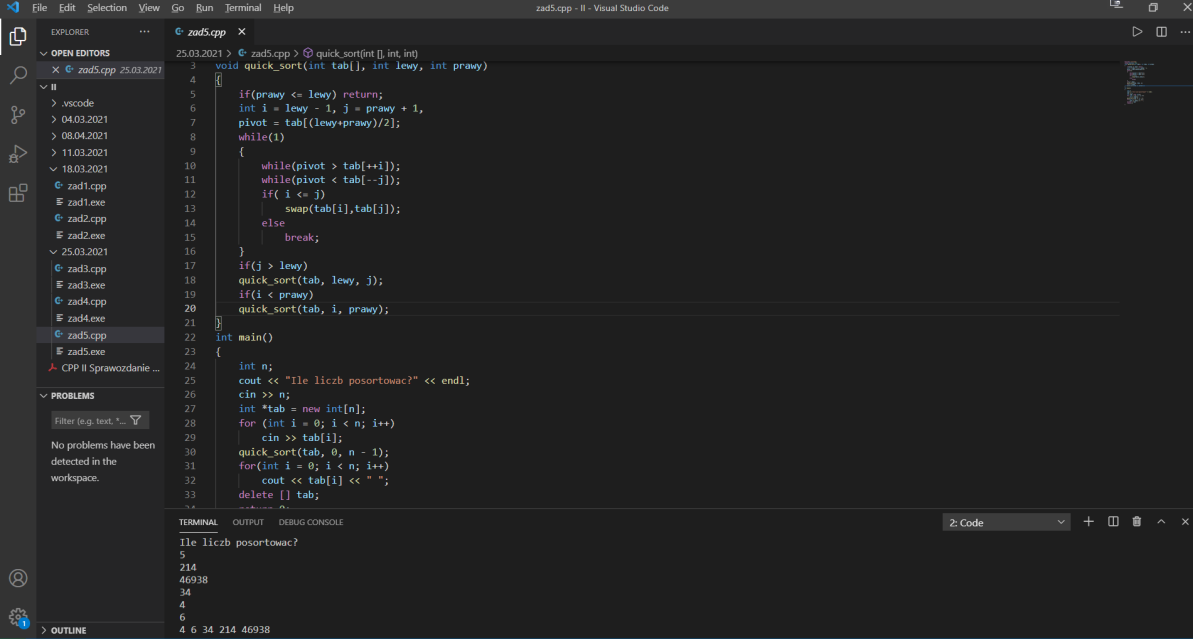
The screenshot shows the Visual Studio Code editor with the file `zad4.cpp` open. The code implements a binary search algorithm to find the index of a target value `x` in a sorted array `tab`. The `main` function generates a random array, sorts it, and then performs the binary search.

```
4 using namespace std;
5 int binarySearch(int tab[], int l, int r, int x);
6 int main()
7 {
8     int x;
9     int tab[5];
10    srand(time(NULL));
11    for (int &x : tab)
12        x = rand() % 100;
13    sort(tab, tab + 5);
14    for (int &x : tab)
15        cout << x << " ";
16    cout << endl;
17    cout << "Jaka liczbe wyszukac? ";
18    cin >> x;
19    int n = sizeof(tab) / sizeof(tab[0]);
20    int res = binarySearch(tab, 0, n - 1, x);
21    (res == -1) ? cout << "Nie ma takiej liczby w tablicy"
22              : cout << "Szukany element ma indeks " << res;
23    return 0;
24 }
25 int binarySearch(int tab[], int l, int r, int x)
26 {
27     if (r >= 1)
28     {
29         int mid = (l + r) / 2;
30         if (tab[mid] == x) return mid;
31         if (tab[mid] > x) return binarySearch(tab, l, mid - 1, x);
32         return binarySearch(tab, mid + 1, r, x);
33     }
34     return -1;
35 }
```

The terminal output shows the execution of the program:

```
18 61 69 83 95
Jaka liczbe wyszukac? 61
Szukany element ma indeks 1
PS C:\Users\Szymo\Desktop\CPP\II\25.03.2021> cd "C:\Users\Szymo\Desktop\CPP\II\25.03.2021" ; if ($?) { g++ zad4.cpp -o zad4 } ; if ($?) { .\zad4 }
12 20 35 67 81
Jaka liczbe wyszukac? 35
Szukany element ma indeks 2
PS C:\Users\Szymo\Desktop\CPP\II\25.03.2021>
```

5. Napisz program, który zaimplementuje algorytm sortowania szybkiego.



```
File Edit Selection View Go Run Terminal Help
zad5.cpp - Visual Studio Code

EXPLORER
  OPEN EDITORS
    zad5.cpp 25.03.2021
  Explorer
    .vscode
    04.03.2021
    08.04.2021
    11.03.2021
    18.03.2021
    zad1.cpp
    zad1.exe
    zad2.cpp
    zad2.exe
    25.03.2021
    zad3.cpp
    zad3.exe
    zad4.cpp
    zad4.exe
    zad5.cpp
    zad5.exe
    CPP II Sprawozdanie ...
  PROBLEMS
    Filter (e.g. text, ...)
    No problems have been detected in the workspace.
  OUTLINE
    4 6 34 214 46938

zad5.cpp
3 void quick_sort(int tab[], int lewy, int prawy)
4 {
5     if(prawy <= lewy) return;
6     int i = lewy - 1, j = prawy + 1,
7     pivot = tab[(lewy+prawy)/2];
8     while(1)
9     {
10         while(pivot > tab[++i]);
11         while(pivot < tab[--j]);
12         if(i <= j)
13             swap(tab[i], tab[j]);
14         else
15             break;
16     }
17     if(j > lewy)
18         quick_sort(tab, lewy, j);
19     if(i < prawy)
20         quick_sort(tab, i, prawy);
21 }
22 int main()
23 {
24     int n;
25     cout << "Ile liczb posortowac?" << endl;
26     cin >> n;
27     int *tab = new int[n];
28     for (int i = 0; i < n; i++)
29         cin >> tab[i];
30     quick_sort(tab, 0, n - 1);
31     for(int i = 0; i < n; i++)
32         cout << tab[i] << " ";
33     delete [] tab;
34     return 0;
35 }
```

TERMINAL OUTPUT DEBUG CONSOLE

```
Ile liczb posortowac?
5
214
46938
34
4
6
```

Ln 20, Col 31 Tab Size: 4 UTF-8 CRLF C++ Win32