
W pliku `liczby.txt` zapisano 500 liczb całkowitych dodatnich po jednej w każdym wierszu. Każda liczba jest z zakresu od 1 do 100 000. Napisz program(-y) dający(-e) odpowiedzi do poniższych zadań. Zapisz uzyskane odpowiedzi w pliku `wyniki4.txt`, poprzedzając każdą z nich numerem odpowiedniego zadania.

Uwaga: Plik `przyklad.txt` zawiera przykładowe dane spełniające warunki zadania. Odpowiedzi dla danych z tego pliku są podane pod treściami zadań.

ZADANIE 2.1

Podaj, ile z podanych liczb jest potęgami liczby 3 (czyli liczbami postaci $1 = 3^0$, $3 = 3^1$, $9 = 3^2$ itd.).

Dla pliku `przyklad.txt` odpowiedź wynosi 2.

```
#include <iostream>
#include <fstream>
#include <cmath>
using namespace std;
int main()
{
    int liczby[501];
    int ilosc=0;
    ifstream odczyt("przyklad.txt");
    for (int i=0; i<500; i++)
        odczyt>>liczby[i];
    odczyt.close();
    for (int i=0; i<500; i++)
    {
        double wyk=log(liczby[i])/log(3);
        if (wyk==(int)wyk)
            ilosc++;
    }
    cout<<"ilosc liczb bedacych potega liczby 3: "<<ilosc;
    return 0;
}
```

ZADANIE 2.2

Silnią liczby naturalnej k większej od 0 nazywamy wartość iloczynu $1 \cdot 2 \cdot \dots \cdot k$ i oznaczamy przez $k!$.

Przyjmujemy, że $0! = 1$. Zatem mamy:

$$0! = 1,$$

$$1! = 1,$$

$$2! = 1 \cdot 2 = 2,$$

$$3! = 1 \cdot 2 \cdot 3 = 6,$$

$$4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24 \text{ itd.}$$

Dowolną liczbę naturalną możemy rozbić na cyfry, a następnie policzyć sumę silni jej cyfr. Na przykład dla liczby 343 mamy $3! + 4! + 3! = 6 + 24 + 6 = 36$.

Podaj, w kolejności ich występowania w pliku `liczby.txt`, wszystkie liczby, które są równe sumie silni swoich cyfr.

W pliku `przyklad.txt` znajduje się jedna taka liczba: 145 ($1! + 4! + 5! = 1 + 24 + 120 = 145$).

```

#include <string>
#include <iostream>
#include <fstream>
using namespace std;
int silnia(int n)
{
    if (n==0)
        return 1;
    return silnia(n-1)*n;
}
int main()
{
    int liczba[501];
    ifstream odczyt("liczby.txt");
    for (int i=0; i<500; i++)
        odczyt>>liczba[i];
    odczyt.close();
    for (int i=0; i<500; i++)
    {
        int suma=0;
        int cyfra=0;
        int a=liczba[i];
        while (a!=0)

```

```

    {
        cyfra=a%10;
        a/=10;
        suma+=silnia(cyfra);
    }
    if (suma==liczba[i])
        cout<<liczba[i]<<endl;
}
return 0;
}

```

ZADANIE 2.3

W pliku `liczby.txt` znajdź najdłuższy ciąg liczb występujących kolejno po sobie i taki, że największy wspólny dzielnik ich wszystkich jest większy od 1 (innymi słowy: istnieje taka liczba całkowita większa od 1, która jest dzielnikiem każdej z tych liczb).

Jako odpowiedź podaj wartość pierwszej liczby w takim ciągu, długość ciągu oraz największą liczbę całkowitą, która jest dzielnikiem każdej liczby w tym ciągu. W pliku z danymi jest tylko jeden taki ciąg o największej długości.

Uwaga: Możesz skorzystać z zależności $\text{NWD}(a, b, c) = \text{NWD}(\text{NWD}(a, b), c)$.

Przykład:

Dla liczb 3, 7, 4, 6, 10, 2, 5 odpowiedzią jest 4 (pierwsza liczba ciągu), 4 (długość ciągu) i 2 (największy wspólny dzielnik), natomiast dla liczb 5, 70, 28, 42, 98, 1 odpowiedzią jest 70 (pierwsza liczba ciągu), 4 (długość ciągu) i 14 (największy wspólny dzielnik).

Odpowiedź dla pliku `przyklad.txt`: pierwsza liczba ciągu 90, długość 5, największy wspólny dzielnik 10.