

## Otwarcie pliku w trybie do odczytu

Wszystkie funkcje (metody) oraz zmienne (atrybuty) zdefiniowane są w klasie o nazwie `ifstream`, na podstawie której musimy utworzyć obiekt. Czytanie zawartości pliku można utożsamić z czytaniem danych z klawiatury. Należy tylko zmienić standardowe wejście z klawiatury na plik. Popatrzmy jeszcze raz, w jaki sposób czytamy dane z klawiatury:

```
cin>>zmienna;
```

gdzie `cin` jest obiektem, który ma zdefiniowane niezbędne metody i atrybuty potrzebne do czytania z klawiatury.

**Czytanie pliku działa na takiej samej zasadzie.** Różnica jest taka, że zamiast nazwy `cin` wpisujemy własną, którą stworzymy w następujący sposób:

```
ifstream jakas_nazwa; //zamiast cin będzie jakas_nazwa
```

```
jakas_nazwa.open("ścieżka do pliku (względna lub bezwzględna)");
```

Klasa `ifstream` ma zdefiniowane wszystkie elementy pozwalające czytać plik.

Jak mamy już obiekt, to możemy skorzystać z pierwszej jego funkcji, która określa fizyczny plik na dysku. W cudzysłowie podajemy ścieżkę do niego bezwzględną lub względną: bezwzględna ścieżka to cały adres do pliku np.: `c:/dane/dane.txt`, względna, to taka, która określa położenie pliku względem programu, np. jeśli program i plik z danymi znajdują się w tym samym miejscu, to wystarczy zapisać

```
jakas_nazwa.open("dane.txt");
```

Plik, który chcemy czytać, możemy zdefiniować od razu przy tworzeniu obiektu i pomijamy w tej sytuacji funkcję `open`

```
ifstream jakas_nazwa("ścieżka do pliku");
```

Teraz, dane wczytujemy za pomocą konstrukcji:

```
jakas_nazwa>>zmienna;
```

Przykład:

Założmy, że mamy utworzony plik tekstowy dane.txt z zapisanymi w nim dwiema liczbami np. 23 i 12. Chcemy odczytać te liczby i wykonać na nich działanie dodawania. **Dodatkowo, nasz program wykonywalny jest w tym samym folderze co plik dane.txt.**

```
#include<iostream>
#include<fstream>

using namespace std;

int main()
{
    int a, b;
    ifstream odczyt("dane.txt");

    odczyt>>a>>b;

    cout<<"Suma wczytanych liczb wynosi: "<<a+b;

    odczyt.close(); //zamknięcie pliku

    return 0;
}
```

Do zamykania pliku służy funkcja **close()**;

Pamiętajmy, że jeśli jakiś plik otwieramy, to po zakończonej pracy musimy go zamknąć!!

## Czytanie całego pliku

Cały plik możemy przeczytać na kilka sposobów.

**\*\*\* Pierwszy sposób to czytanie pliku znak po znaku:**

```
#include<iostream>
#include<fstream>

using namespace std;

int main()
{
    int a, b;
    ifstream odczyt("dane.txt");

    if(odczyt.is_open())
    {
        char znak;
        while(odczyt>>znak) //dopóki jest co czytać
        {
            cout<<znak; //wypisz to co wczytałeś z pliku
            //lub wykonaj inną operację
        }
    }
}
```

```
    else
        cout<<"Nie udało się otworzyć pliku";

    return 0;
}
```

### \*\*\* drugi sposób to czytanie **wyraz po wyrazie**:

```
#include<iostream>
#include<fstream>

using namespace std;

int main()
{
    int a, b;
    ifstream odczyt("dane.txt");

    if(odczyt.is_open())
    {
        char wyraz[100]; //maksymalnie 99 znaków
        while(odczyt>>wyraz) //dopóki jest co czytać
        {
            cout<<wyraz<<" "; //wypisz to co wczytałeś z pliku
            //lub wykonaj inną operację
        }
    }
    else
        cout<<"Nie udało się otworzyć pliku";

    return 0;
}
```

### \*\*\* trzeci sposób **wiersz po wierszu**:

```
#include<iostream>
#include<fstream>

using namespace std;

int main()
{
    int a, b;
    ifstream odczyt("dane.txt");

    if(odczyt.is_open())
    {
        char wiersz[10000]; //maksymalnie 9999 znaków w wierszu
```

```

        while(odczyt.getline(wiersz,10000)) //dopóki jest co czytać
        {
            cout<<wiersz<<endl; //wypisz to co wczytałeś z pliku
            //lub wykonaj inną operację
        }
    }
    else
        cout<<"Nie udało się otworzyć pliku";

    return 0;
}

```

Zapis do pliku:

*Otwarcie pliku w trybie do zapisu*

Do czytania pliku musimy stworzyć obiekt na podstawie klasy **ofstream**

, która przechowuje wszystkie metody i atrybuty operujące na plikach do zapisu. W tym przypadku możemy dopatrzeć się analogii takiej jak w wyświetlaniu danych na ekranie. W tym celu użyjemy konstrukcji

```
cout<<dane;
```

Zamiast obiektu **cout** użyjemy własnego obiektu stworzonego na bazie klasy **ofstream**

```
ofstream jakas_nazwa("ścieżka do pliku");
```

lub:

```
ofstream jakas_nazwa;
```

```
jakas_nazwa.open("ścieżka do pliku");
```

Zapis informacji wygląda następująco:

```
ofstream jakas_nazwa("ścieżka do pliku");
```

```
    jakas_nazwa<<informacja;
```

Należy pamiętać, że jeśli plik nie istnieje to **zostanie on utworzony**, natomiast jeśli istnieje i są w nim już zapisane jakieś informacje, to zostaną one **nadpisane**.

## Przykład:

Napiszemy program, który wczyta z klawiatury dwie liczby całkowite i zapisze je wraz z ich sumą w postaci:

$$\text{pierwsza\_liczba} + \text{druga\_liczba} = \text{suma}$$

do pliku *dane.txt*.

```
#include<iostream>
#include<fstream>

using namespace std;

int main()
{
    int a, b;
    cout<<"Podaj dwie liczby: ";
    cin>>a>>b;

    ofstream zapis("dane.txt");

    zapis<<a<<" + "<<b<<" = "<<a+b;

    zapis.close(); //obowiązkowo należy zamknąć plik

    return 0;
}
```

Jeśli podamy 5 i 10 to w pliku *dane.txt* pojawi się taki ciąg: 5 + 10 = 15

## Dopisywanie informacji do pliku

Jeśli chcemy dopisać dane na koniec pliku to musimy to zadeklarować przy jego otwarciu ustawiając tryb na `ios::base::app`

```
ofstream jakas_nazwa("ścieżka", ios_base::app);
#include<iostream>
#include<fstream>

using namespace std;

int main()
{
    char *tekst = "Ala ma kota\n";

    ofstream zapis("dane.txt");//obiekt tworzymy tylko raz
    zapis<<tekst;
    zapis.close(); //obowiązkowo należy zamknąć plik

    zapis.open("dane.txt", ios_base::app);
    zapis<<tekst;
    zapis.close();

    return 0;
}
```

```
}
```

## Czy udało się otworzyć plik?

Przed wykonaniem jakiegokolwiek operacji na pliku, warto upewnić się, że został on prawidłowo otwarty. Plik może zostać nieotwarty z wielu powodów np. jeśli:

- podamy nieprawidłową ścieżkę do pliku
- plik jest używany przez inny program
- brak dostępu
- plik nie istnieje
- itp.

Do skontrolowania tej czynności służy metoda o nazwie *is\_open()*. Zwraca ona **true**, jeśli plik został otwarty i **false** w przeciwnym razie.

### Przykład.

```
#include<iostream>
#include<fstream>

using namespace std;

int main()
{
    int a, b;
    ifstream odczyt("dane.txt");

    if(odczyt.is_open())
    {
        odczyt>>a>>b;
        cout<<"Suma wczytanych liczb wynosi: "<<a+b;
        cin.get();
        odczyt.close(); //zamknięcie pliku
    }
    else
        cout<<"Nie udało się otworzyć pliku";

    return 0;
}
```

### Ćwiczenie:

1. Wygeneruj do pliku o nazwie dane.txt 100 liczb z dowolnego przedziału
2. Wczytaj wygenerowany plik do programu i następnie podnieś liczby do kwadratu i zapisz plik o nazwie wynik.txt.