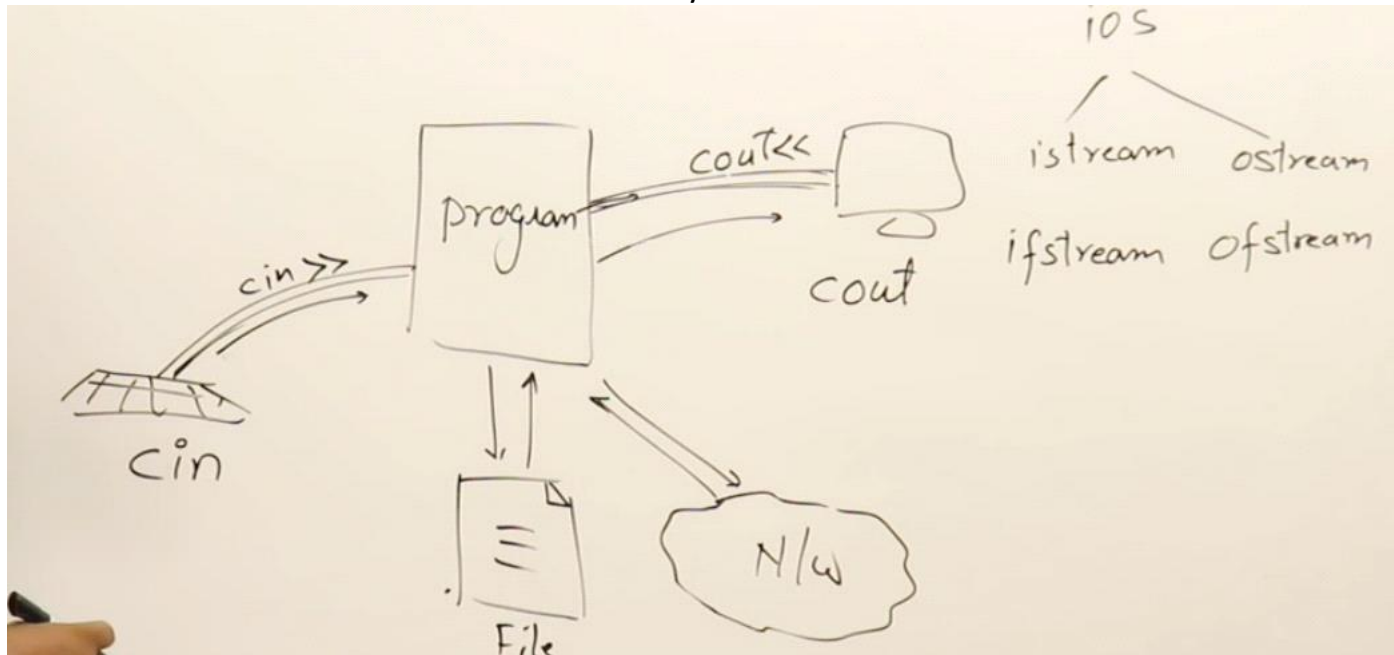


# IO Streams

24 September 2024 19:55

Streams are flow of character or should we say flow of data.



## File Handling

It involves reading to writing to files using streams. The standard Library provides :

- Ifstream : for input file handling
- Ofstream : for output file handling
- Fstream : for both input and output file handling

```

#include<iostream>
#include<fstream> // contains the classes ifstream, ofstream, and fstream

using namespace std;

int main() {
    // Opening a file for reading
    ifstream inputFile;
    inputFile.open("input.txt"); // Use double quotes for file names
    if (!inputFile) {
        cerr << "File couldn't be opened!" << endl;
        return 1;
    }

    // Opening a file for writing
    ofstream outputFile("output.txt");
    if (!outputFile) {
        cerr << "Output file couldn't be opened!" << endl;
        return 1;
    }

    // Reading from input.txt and writing to output.txt
    string line;
    while (getline(inputFile, line)) {
        cout << line << endl; // Displaying the content of input.txt
        outputFile << line << endl; // Writing content to output.txt
    }


    // Close both files after operation
    inputFile.close();
    outputFile.close();

    cout << "File operation completed!" << endl;
    return 0;
}

```

### Reading Word by Word:

cpp

 Copy code


```
std::ifstream inputFile("input.txt");
std::string word;

while (inputFile >> word) {
    std::cout << word << std::endl;
}

inputFile.close();
```

### Reading Character by Character:

cpp

 Copy code

```
std::ifstream inputFile("input.txt");
char ch;

while (inputFile.get(ch)) {
    std::cout << ch;
}

inputFile.close();
```



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

using namespace std;

int main(){
    fstream file;
    file.open("input.txt", ios::in | ios::out | ios::app); // both input output
    if (!file){
        cerr << "File couldn't be opened!" << endl;
        return 1;
    }

    // Write to the file
    file << "Appending this line to the file." << endl;

    // Move file pointer back to the beginning
    file.seekg(0);

    // Read the file
    string line;
    while (getline(file, line)){
        cout << line << endl;
    }

    file.close();
    return 0;
}
```