

The C++ logo is displayed in white on a black rectangular background. It consists of a large 'C' followed by two '+' signs.

Input & Output
Console & File

SHORT GUIDE

-Basic
structure
-Declare
variables
-Input &
output from
console
-Input &
output from
file

```
#include <iostream>
#include <fstream>
using namespace std;

int main() {
    ifstream fin ("input.in");
    ofstream fout ("output.out");

    int a;
    cin >> a;
    cout << a;

    int b;
    fin >> b;
    fout << b;

    return 0;
}
```

Read from file a known number of values

Example: Read 'number' names and the prices. 'Number' is in the first line of the file.

Sample Input 1

```
3
Ahmed
300
Suzanne
500
Ivona
450
```

```
# include <fstream>
# include <iostream>
using namespace std;

int main(){
    ifstream fin ("input.in");
    ofstream fout ("output.out");

    int number;
    fin >> number;    // input first line of file as number
    string name;     // string to keep the names
    int price;       // integer to keep the prices

    // for each line of file:
    for (int i=0; i<number; i++){
        fin >> name;
        fin >> price;
        // ...more code...
    }

    return 0;
}
```

Read from file an unknown number of values

Example: Read values until
you get 99999.

Sample Input

```
57234
00907
34100
99999
```

```
# include <fstream>
# include <iostream>
using namespace std;

int main(){
    ifstream fin ("input.in");
    ofstream fout ("output.out");

    int x;
    while(fin >> x)
        if (x != 99999) cout << x;
    }

return 0;
}
```

Keep a known number of values in array

Example: Read pairs of values.
The number of pairs is given in the first line.

Sample Input

```
5
44, 62
34, 69
24, 78
42, 44
64, 10
```

```
# include <fstream>
# include <iostream>
using namespace std;
int main(){
    ifstream fin ("input.in");
    ofstream fout ("output.out");
    int number;
    fin >> number;
    cout << "number is "<<number<<endl;
    int xcor[number];
    int ycor[number];
    int x;
    int y;
    char ch;
    for (int i=0; i<number; i++){
        fin >> x;           //input first number
        fin >> ch;          //input comma
        fin >> y;           //input second number
        xcor[i] = x;       //assign number to array
        ycor[i] = y;       //assign number to array
    }
    return 0;
}
```

Keep a known number of values in array (II)

Example: Read pairs of values.
The number of pairs is given in the first line.

Sample Input

```
5
44, 62
34, 69
24, 78
42, 44
64, 10
```

```
# include <fstream>
# include <iostream>
using namespace std;
//If maximum number of lines is known
//you can declare the array with maximum size
//Let's suppose that maximum number is 100000
//declare arrays outside the main as global variable
int xcor[100005];    //declare an array a little bit bigger
int ycor[100005];
int main(){
    ifstream fin ("input.in");
    ofstream fout ("output.out");
    int number;
    fin >> number;
    cout << "number is " << number << endl;
    int x;
    int y;
    char ch;
    for (int i=0; i<number; i++){
        fin >> x;           //input first number
        fin >> ch;         //input comma
        fin >> y;           //input second number
        xcor[i] = x;       //assign number to array
        ycor[i] = y;       //assign number to array
    }
    return 0;
}
```

Read a string from file and access its characters

Example: Read two strings

Sample Input 1

ABCCDEABAA

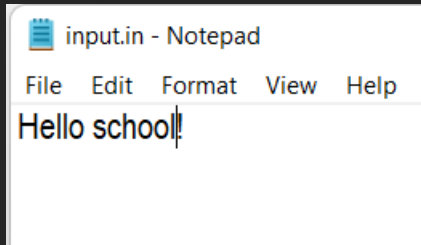
ABCDE

```
# include <fstream>
# include <iostream>
using namespace std;

int main(){
    ifstream fin ("input.in");
    ofstream fout ("output.out");
    string S1;
    string S2;
    fin >> S1;
    fin >> S2;
    cout << S1<<endl;
    cout << S2<<endl;
    for (int i=0;i<S1.length();i++) {
        cout << S1[i]<<endl;
    }
    return 0;
}
```

Read a line from file

Example: Read line and print to console



```
Hello school!  
Process returned 0 (0x0)   execution time : 0.030 s  
Press any key to continue.
```

```
# include <fstream>  
# include <iostream>  
using namespace std;  
  
int main() {  
    ifstream fin ("input.in");  
    ofstream fout ("output.out");  
    string phrase;  
    getline(fin, phrase);  
    cout << phrase;  
    return 0;  
}
```