

* **Purpose** : Classwork.

* **Date** : 05/01/2026

* **Author** : Vikas Srivastava

* **ID** : 55984

* **Batch ID** : 25SUB4505

```
C cmdLineArgs.c X
1
2 // Purpose - Write a program to display command-line arguments in C, showing the count of arguments (argc) and listing each argument (argv).
3
4 #include <stdio.h>
5
6 int main(int argc, char *argv[]){
7     printf("Count of args: %d\n", argc);
8     printf("Argument List: \n");
9     for(int cnt = 0; cnt < argc; cnt++){
10         printf("\targv[%d] ==> %s\n", cnt, argv[cnt]);
11     }
12 }
```

Count of args: 1
Argument List:
argv[0] ==> c:\Users\VIKAS SRIVASTAVA\OneDrive\Desktop\C_CPP\Day_10\Classwork\cmdLineArgs

```
C fileBinary.cpp X
1
2 // Purpose - Write a program to read a binary file into an object of a class in C++ and display the object's data, demonstrating default initialization and file input using ifstream.
3
4 #include <iostream>
5 #include <string>
6 #include <fstream>
7 using namespace std;
8
9 class Employee{
10     int id;
11     char name[40];
12     double sal;
13 public:
14     Employee(int num=1001, const char *na="Sample #1", double sa = 15000.05):id(num), sal(sa){
15         strcpy(name, na);
16     }
17     void disp(){
18         cout<<"Output Num: "<<id<<"(Name: "<<name<<"\tSal: "<<sal<<endl;
19     }
20 };
21
22 int main(){
23     Employee emp1;
24
25     emp1.disp(); //default values
26
27     ifstream fobj("mydata.dat", ios::binary);
28
29     if (fobj){
30         fobj.read((char *) &emp1, sizeof(emp1));
31         emp1.disp();
32     }
33 }
```

Output Num: 1001 Name: Sample #1 Sal: 15000
Output Num: 1101 Name: Karthik Sal: 15000.5

```
C fileLine.cpp X
1
2 // Purpose - Write a program to read a text file line by line in C++ and display its contents on the console using ifstream and getline().
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 int main(){
9     ifstream fobj("iothree.cpp");
10     string str;
11     while (getline(fobj, str)) //word by word
12         cout<<str<<endl;
13
14 }
```

```

1 // Purpose - Write a program to open a text file in C++ with error checking, read it line by line, and display its contents on the console.
2
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 int main(){
9     char fileName[] = "iothree.txt";
10    ifstream fobj(fileName);
11    string str;
12    if (!fobj){
13        cout<<fileName<<" file NOT FOUND"<<endl;
14        return 1;
15    }
16
17    while (getline(fobj,str)) //word by word
18        cout<<str<<endl;
19
20 }

```

ioThree.txt file NOT FOUND

```

1 // Purpose - Write a program to write data to a text file in C++ using ofstream.
2
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 int main(){
9     ofstream fobj("fileOut.txt");
10    fobj<<"Sending data into the file"<<endl;
11 }

```

```

1 // Purpose - Write a program to append data to an existing text file in C++ using ofstream with ios::app mode.
2
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 int main(){
9     ofstream fobj("fileOut.txt", ios::app);
10    fobj<<"Sending data into the file"<<endl;
11 }

```

```

1 // Purpose - Write a program to write an object of a class to a binary file in C++, demonstrating binary file output using ofstream.
2
3
4 #include <iostream>
5 #include <fstream>
6 #include <string>
7 using namespace std;
8
9 class Employee{
10     int id;
11     char name[50];
12     double sal;
13 public:
14     Employee(int num=1001, const char *na="Sample #1", double sa = 15000.05):id(num), sal(sa){
15         strcpy(name, na);
16     }
17     void disp(){
18         cout<<"Output Num: "<<id<<"\tName: "<<name<<"\tSal: "<<sal<<endl;
19     }
20 };
21
22 int main(){
23     Employee emp1 = Employee(1101,"Karthik", 15000.45);
24
25     ofstream fobj("myData.dat", ios::binary);
26
27     fobj.write((char *) &emp1, sizeof(emp1));
28     emp1.disp();
29 }
30

```

Output Num: 1101 Name: Karthik Sal: 15000.5

```

1 // Purpose - Write a program to demonstrate the use of tellp() in C++, showing how to get the current write position in an output file stream.
2
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 int main() {
9     ofstream fout("dataTest.txt");
10
11     cout << "Initial position: " << fout.tellp() << endl;
12
13     fout << "Hello";
14     cout << "After writing 'Hello': " << fout.tellp() << endl;
15
16     fout << " World";
17     cout << "After writing ' World': " << fout.tellp() << endl;
18
19     fout.close();
20 }

```

```

Initial position: 0
After writing 'Hello': 5
After writing ' World': 11

```

```

1 // Purpose - Write a program to read a text file word by word in C++ using ifstream and the extraction operator (>>) and display the words on the console.
2
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 int main(){
9     ifstream fobj("iothree.cpp");
10    string str;
11    while (fobj>>str) //word by word
12        cout<<str<<" ";
13
14 }

```

```

1 // Purpose - Write a program to read input character by character from the standard input in C++ and immediately echo it to the console.
2
3
4 #include <iostream>
5 using namespace std;
6
7 int main(){
8     char ch;
9     while ( cin >> ch)
10         cout<<ch;
11
12 }

```

```

1 // Purpose - Write a program to read a text file character by character in C++ using ifstream::get() and display its contents on the console.
2
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 int main(){
9     ifstream fobj("iothree.cpp");
10    char ch;
11    while ( fobj.get(ch)
12        cout<<ch;
13
14 }

```

```

1 // Purpose - Write a program to display the contents of a text file in C++, optionally numbering each line when a -n command-line option is provided.
2
3
4 #include <iostream>
5 #include <fstream>
6 using namespace std;
7
8 void fileOut(istream& in, ostream& out, bool lineNu=false);
9
10 int main(int argc, char *argv[]){
11     if (argc == 2){
12         ifstream fobj(argv[1]);
13         fileOut(fobj, cout);
14     }else if ((argc == 3) && (argv[1][0] == '-') && (argv[1][1] == 'n')){
15         ifstream fobj(argv[2]);
16         fileOut(fobj, cout, true);
17     }
18 }
19
20 void fileOut(istream& in, ostream& out, bool lineNu){
21     char ch;
22     int num = 1;
23     if(lineNu)
24         cout<<num<<endl;
25     while(in.get(ch)){
26         out<<ch;
27         if ((ch=='\n') && lineNu)
28             cout<<num<<endl;
29     }
30 }

```