Lab9
Streams and File I/O

DEE 1319
Department of Electronics Engineering
National Chiao Tung University

Outline

- Streams & File I/O
- Lab9 exercise

Streams

- A flow of characters
- Input stream
  - Flow into program
    - From keyboard/file/string
- Output stream
  - Flow out of program
    - Go to screen/file/string
- I/O streams
  - Both input and output directions

Streams Usage

- We've used streams already
  - cin
    Input stream (istream) object connected to stdin (keyboard by default)
  - cout
    Output stream (ostream) object connected to stdout (screen by default)
  - Cerr
    Output stream (ostream) object connected to stderr (screen by default)
- Can define other streams
  - To or from files
  - Used similarly as cin, cout

File Connection

- Must first connect file to stream object
- For input:
  - File ➔ ifstream object
- For output:
  - File ➔ ofstream object
- Classes ifstream and ofstream
  - Defined in library <fstream>
  - Named in std namespace

File I/O Libraries

- To allow both file input and output in your program:

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

OR

```
#include <fstream>
using std::ifstream;
using std::ofstream;
```

File Open and Close

- Two ways for file open
  - Method 1
    ifstream ifs(“pathname\input_file_name”);
    ofstream ofs(“pathname\output_file_name”);
  - Method 2
    ifstream ifs; ifs.open(“pathname\input_file_name”);
    ofstream ofs; ofs.open(“pathname\output_file_name”);
- Close
  - ifs.close();
  - ofs.close();

Streams Usage

- Once intput/output file stream are successfully opened
  ➔ You can use them just like input/output streams

```
int i;
double d;
char str[100];

ifstream ifs("ifilename");
ifs >> i >> d;  ifs.getline(str, 100);  //not from keyboard but input file

ofstream ofs("ofilename");
ofs << i << endl <<setprecision(8) << fixed <<d;
ofs.put("\n");                          //not to screeen but output file
```

Checking File Open Success

- File opens could fail
  - If input file doesn't exist
  - No write permissions to output file
  - Unexpected results
- Use member function fail() or operator !

```
ifstream ifs("in.txt");
if (ifs.fail()){
    cout << "File open failed.\n";
    exit(1);
}

ofstream ofs("out.txt");
if(!ofs){
    cout << "Output file open failed.\n";
    exit(1);
}
```

Checking End of File

- Use loop to process file until end
  - Typical approach
- Test for end of file
  - Member function eof()
  - eof() returns true if end-of-file is reached

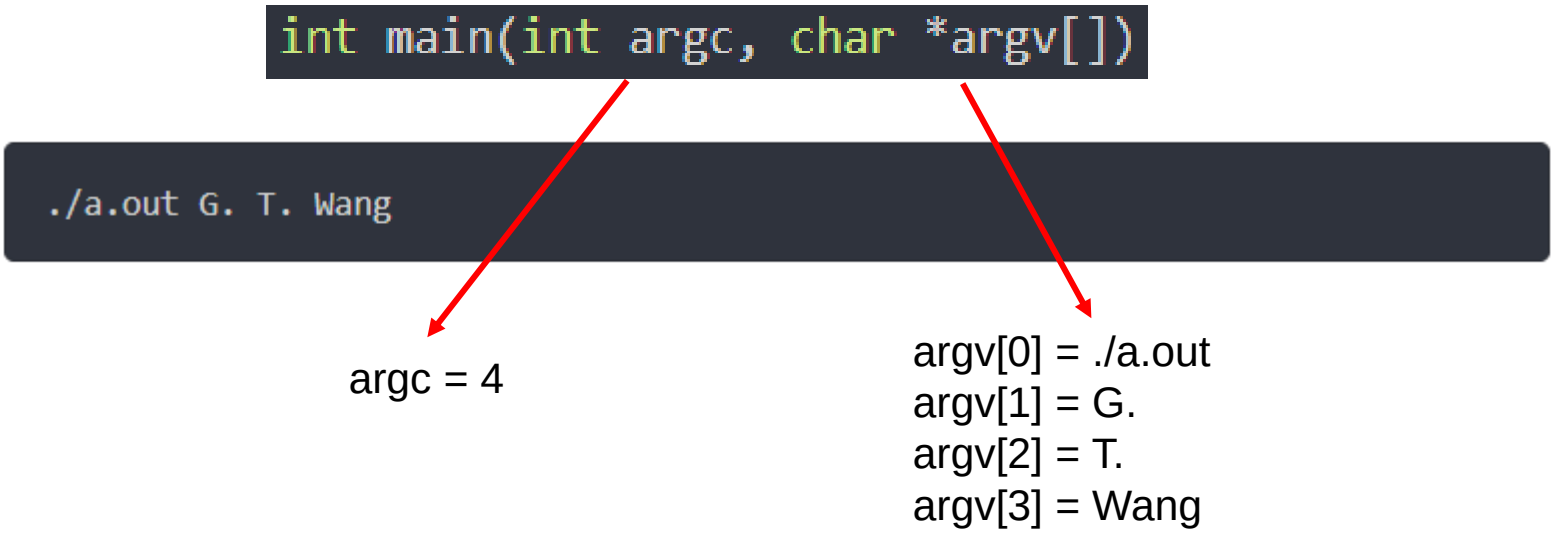
```
char next;
ifs.get(next);
while (!ifs.eof()){
    cout << next;
    ifs.get(next);
}
```

Lab9 Exercise (1/2)

- The program utilizes the following stream for reading input data from a file and writing output data to another file
  - ifstream
    - open an input file stream for reading data from a file
  - ofstream
    - open an output file stream for writing data to a file
  - cerr
    - output error messages to the standard error stream, typically used for error handling purposes

Lab9 Exercise (2/2)

- Input and output file name handling with command-line arguments
  - The program utilizes command-line arguments (argv and argc) to specify input and output file paths
  - provides the path directly from the command line when executing the program



Problem

- You need to reads all data of all students from the input file, including the student's name and the list of classes he/she enrolled in

- Given: an input file with student's information
- Output: an output file with class's information

Input

- Input format
  - <num of students>
    <num of classes>
    <class1> <class2> …….

    <name1> <num of classes> <class> …….
    <name2> <num of classes> <class> …….
    …….
  - Example:

```
5
4
MATH ENGLISH PHYSICS CHEMISTRY

Amy 2 MATH ENGLISH
Ben 3 ENGLISH PHYSICS CHEMISTRY
Cindy 4 MATH ENGLISH PHYSICS CHEMISTRY
David 1 PHYSICS
Emma 2 MATH CHEMISTRY
```

Output

- Output format
  - Class: <classname1>
    <all students enrolled in it>
    Class: <classname2>
    <all students enrolled in it>
    …
  - Example

```
1   Class : CHEMISTRY
2       Ben Cindy Emma
3   Class : ENGLISH
4       Amy Ben Cindy
5   Class : MATH
6       Amy Cindy Emma
7   Class : PHYSICS
8       Ben Cindy David
```

Both the classes and student names are sorted alphabetically (A->Z) !!