Lab 5

C++ File I/O

- C++ file I/O are easy to implement using the class of ofstream and ifstream in the **fstream** library.
- ofstream: File classes for write operations (derived from ostream)
- ifstream: File classes for read operations (derived from istream)
- **fstream**: File classes for both reads and writes (derived from iostream)

C++ File I/O

- File write

  ofstream object_name(“file_name");

- File read

  ifstream object_name(“file_name"); or

  ifstream object_name
  object_name.open(“file_name");

C++ File I/O

- Major function of ifstream/ofstream
  - Constructor (argument is string of filename)
    - File open
  - is_open()
    - When file is open, return True
  - close()
    - File close

C++ File I/O

- getline(ifstream, string)
- getline()은 개행문자(‘ \ n’)를 입력의 끝으로 인식하여 정해진 길이만큼 한 줄 전체를 읽어 들인다.

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

using namespace std;

int main()
{
        string filePath = "test.txt";

        // write File
        ofstream writeFile(filePath.data());
        if( writeFile.is_open() ){
                writeFile << "Hello World!₩n";
                writeFile << "This is C++ File Contents.₩n";
                writeFile.close();
        }

        // read File
        ifstream openFile(filePath.data());
        if( openFile.is_open() ){
                string line;
                while(getline(openFile, line)){
                        cout << line << endl;
                }
                openFile.close();
        }

        return 0;
}
```

Exercise 7

- **Pre-requisite knowledge** before Project 1
- String manipulation(Search, Split, Compare, …)
- File I/O (input only for today)
- No submit on ETL this week.
- Your goal is to show the right result from a text file.
- This week's task is done with cpp.
- Practice recommended with JAVA (java.io.file & java.io.filereader)

From a text file with one string line

- Go over to **Visual Studio Code**
- Make a new file titled “test0412.txt”.
- In the text file, copy the following string and save it.

[R200T25H10],[S2000T20H0A1000D20],[O30T10D500W100]

- Get the string from the file (using ifstream)
- Save that string into a variable which data type is “char []”
- Split the string into the form in the next page.

Sample code (char [] split)

```
#include <iostream>
#include <cstring>
#include <vector>
#include <fstream>

using namespace std;

int main ()
{
	char str[512];
	char * pch;
	const char* delimiter = ",";
	vector<char *> a;

	string filePath = "test0412.txt";

	ifstream openFile(filePath.data());
	if(openFile.is_open()) {
	openFile.getline(str,512);
	openFile.close();
	}
	printf ("Splitting string \"%s\" into tokens:\n",str);
	pch = strtok (str,delimiter);
	while (pch != NULL)
	{
	printf ("%s\n",pch);
	a.push_back(pch);
	pch = strtok (NULL,delimiter);
	}
	return 0;
}
```

```
Splitting string "[R200T25H10],[S2000T20H0A1000D20],[O30T10D500W100]" into tokens:
[R200T25H10]
[S2000T20H0A1000D20]
[O30T10D500W100]
```

Sample code continued

```
#include <iostream>
#include <cstring>
#include <vector>
#include <fstream>

using namespace std;

int main ()
{
	char str[] = "[R100T10H5]";
	char title[] = "RTH";
	int i = 0;

	char * pch;
	const char* delimiter = "[]RTH";
	vector<char *> a;

	printf ("Splitting string \"%s\" into tokens:\n",str);
	pch = strtok (str,delimiter);
	while (pch != NULL)
	{
	printf ("%c: %s\n",title[i++],pch);
	a.push_back(pch);
	pch = strtok (NULL,delimiter);
	}
	return 0;
}
```

```
Splitting string "[R100T10H5]" into tokens:
R: 100
T: 10
H: 5
```

Sample code (Split with string data type)

```cpp
#include <iostream>
#include <sstream>
#include <cstring>
#include <vector>
#include <fstream>

using namespace std;

int main()
{
	string filePath = "test0412.txt";
	string str;
	ifstream openFile(filePath.data());
	if(openFile.is_open()) {
	getline(openFile, str);
	openFile.close();
	}

	stringstream ss(str);
	vector<string> result;

	while( ss.good() )
	{
	string substr;
	getline( ss, substr, ',' );
	result.push_back( substr );
	cout << substr << endl;
	}
	return 0;
}
```

```
[R200T25H10]
[S2000T20H0A1000D20]
[O30T10D500W100]
```

Screenshot of the result

- [R200T25H10],[S2000T20H0A1000D20],[O30T10D500W100]

1. Detect R / S / O on the first of the split block
2. Match each split string to Road / Sky / Ocean, respectively.
3. The program works with the cases when the text shuffles like [S2000….],[O30T….],[R200T….]

4. There is NO limitation of the implementation.
5. This Exercise is for your Project 1.

```
Road Type Environment:
R: 200
T: 25
H: 10

Sky Type Environment:
S: 2000
T: 20
H: 0
A: 1000
D: 20

Ocean Type Environment:
O: 30
T: 10
D: 500
W: 100
```