

Lab #11

Date: 2014/04/30 – Michael Hsu

goo.gl/YZU8gG

YZU8gG

File I/O

Each character stored in a plain-text files has its own position.

There is a special character “end of file”.

Tony 100

Adam 98

T	o	n	y		1	0	0	\n	A	d	a	m		9	8	EOF
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

File streams: #include <fstream>

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

int main()
{
    ifstream inFile("input.txt", ios::in); // Input file streams (read)
    ofstream outFile("output.txt", ios::out); // output file streams (write)

    if(inFile)
    {
        char name[20] = {0};
        char head[30] = {0};
        int score = 0;
        int sumScore = 0;

        inFile.getline(head, 30);

        while(!inFile.eof())
        {
            inFile >> name >> score;
            sumScore += score;
            outFile << name << " ";
        }
        outFile << sumScore << endl;
    }

    inFile.close();
    outFile.close();

    return 0;
}
```

mode	
ios::in	Open for input operations (Default)
ios::out	Open for output operations (Default)
ios::app	Appending the content to the end of the file.
ios::ate	Set the position at the end of file.

File streams: #include <fstream>

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

int main()
{
    ifstream inFile("input.txt", ios::in); // Input file streams (read)
    ofstream outFile("output.txt", ios::out); // output file streams (write)

    if(inFile)
    {
        char name[20] = {0};
        char head[30] = {0};
        int score = 0;
        int sumScore = 0;

        inFile.getline(head, 30);

        while(!inFile.eof())
        {
            inFile >> name >> score;
            sumScore += score;
            outFile << name << " ";
        }
        outFile << sumScore << endl;

        inFile.close();
        outFile.close();

        return 0;
    }
}
```

fstream	getline(char* s, streamsize n, char delim)
s	Character array
n	size
delim	'\n' (Default)

File streams: #include <fstream>

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

int main()
{
    ifstream inFile("input.txt", ios::in); // Input file streams (read)
    ofstream outFile("output.txt", ios::out); // output file streams (write)

    if(inFile)
    {
        char name[20] = {0};
        char head[30] = {0};
        int score = 0;
        int sumScore = 0;

        inFile.getline(head, 30);

        while(!inFile.eof())
        {
            inFile >> name >> score;
            sumScore += score;
            outFile << name << " ";
        }
        outFile << sumScore << endl;

        inFile.close();
        outFile.close();

        return 0;
    }
}
```

input.txt

name score
Tony 100
Adam 98
Robin 95
John 90

EOF

output.txt

Tony Adam Robin John 383

C++ Strings

C string: character array

C++ string: C++ string, `#include <string>`

T	h	i	s		i	s		a	n		a	p	p	l	e	\0			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			

```

#include <iostream>
#include <string>

using namespace std;

int main()
{
    string s1 = "This is ";
    string s2 = "an apple";
    string s = s1 + s2;
    cout << s << endl;

    s.replace(s.find("ap"), 2, "sam"); // 11, 12
    cout << s << endl;

    s.erase(s.begin()+9);
    cout << s << endl;

    s.insert(s.length(), "."); // 16
    cout << s << endl;

    cout << s.compare("This is a sample.") << endl;
    cout << s.compare("This is an apple.") << endl;
    cout << s.compare("There are two apples.") << endl;

    return 0;
}

```

string	replace(size_t pos, size_t len, const string& str)
pos	position (from)
len	length (to)
str	replace string

```

This is an apple
This is an sample
This is a sample
This is a sample.
0
-1
1

```


T	h	i	s		i	s		a	n		a	p	p	l	e	\0			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			

```

#include <iostream>
#include <string>

using namespace std;

int main()
{
    string s1 = "This is ";
    string s2 = "an apple";
    string s = s1 + s2;
    cout << s << endl;

    s.replace(s.find("ap"), 2, "sam"); // 11, 12
    cout << s << endl;

    s.erase(s.begin()+9);
    cout << s << endl;

    s.insert(s.length(), "."); // 16
    cout << s << endl;

    cout << s.compare("This is a sample.") << endl;
    cout << s.compare("This is an apple.") << endl;
    cout << s.compare("There are two apples.") << endl;

    return 0;
}

```

string	find(const string& str, size_t pos)
str	search string
pos	The first position to search (0 Default)

```

This is an apple
This is an sample
This is a sample
This is a sample.
0
-1
1

```

T	h	i	s		i	s		a	n		a	p	p	l	e	\0			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			

```
#include <iostream>
#include <string>

using namespace std;

int main()
{
    string s1 = "This is ";
    string s2 = "an apple";
    string s = s1 + s2;
    cout << s << endl;

    s.replace(s.find("ap"), 2, "sam"); // 11, 12
    cout << s << endl;

    s.erase(s.begin()+9);
    cout << s << endl;

    s.insert(s.length(), "."); // 16
    cout << s << endl;

    cout << s.compare("This is a s");
    cout << s.compare("This is an");
    cout << s.compare("There are t");

    return 0;
}
```

string	
erase	Erase part of string

```
This is an apple
This is an sample
This is a sample
This is a sample.
0
-1
1
```

string	insert(size_t pos, const string& str)
pos	Insert at position pos
str	Insert string

T	h	i	s		i	s		a	n		a	p	p	l	e	\0			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			

```
#include <iostream>
#include <string>

using namespace std;

int main()
{
    string s1 = "This is ";
    string s2 = "an apple";
    string s = s1 + s2;
    cout << s << endl;

    s.replace(s.find("ap"), 2,
    cout << s << endl;

    s.erase(s.begin()+9);
    cout << s << endl;

    s.insert(s.length(), ".");
    cout << s << endl;

    cout << s.compare("This is a sample.") << endl;
    cout << s.compare("This is an apple.") << endl;
    cout << s.compare("There are two apples.") << endl;

    return 0;
}
```

value	String s1.compare(s2)
= 0	s1 equal s2
< 0	Not match, s1 lower than s2
> 0	Not match, s1 greater than s2

0
-1
1

T	h	i	s		i	s		a	n		a	p	p	l	e	\0			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			

```

#include <iostream>
#include <string>

using namespace std;

int main()
{
    string s1 = "This is ";
    string s2 = "an apple";
    string s = s1 + s2;
    cout << s << endl;

    s.replace(s.find("ap"), 2, "sam"); // 11, 12
    cout << s << endl;

    s.erase(s.begin()+9);
    cout << s << endl;

    s.insert(s.length(), "."); // 16
    cout << s << endl;

    cout << s.compare("This is a sample.") << endl;
    cout << s.compare("This is an apple.") << endl;
    cout << s.compare("There are two apples.") << endl;

    return 0;
}

```

```

This is an apple
This is an sample
This is a sample
This is a sample.
0
-1
1

```

More about C++ String class

<http://www.cplusplus.com/reference/string/string/>

Lab Work: Palindrome

Description

Write a program that repeatedly reads words from a input text file “input.txt”. In the input text file, there will be several line contain one word. Please check if the word is palindrome or not. If the word is not palindrome, return 0. If the word is palindrome, separate the word by a symbol “|” in the center. Finally, saving the results to the “output.txt” file.

Sample input (input.txt)

civic

redder

pineapple

Sample output (output.txt)

ci|ic

red|der

0

Q&A