Class 23

Strings and Characters
Text in C++

• There are three types:
  • char – stores a single character
  • string – stores a sequence of characters
  • char[] – also known as c-strings; stores sequence of characters
Characters

• In C++ characters are actually stored as numbers
• Every character is understood as a number using the ASCII code

‘A’ = 65  ‘a’ = 97
‘B’ = 66  ‘b’ = 98
‘C’ = 67  ‘c’ = 99

• https://www.asciitable.com
Type Casting

• `cout << (int) 'F' << endl;  // 70`
• `int i = 80;`
• `cout << (char) i;  // P`
Example 1

• Write function to print a table listing the ASCII codes and characters for ASCII 33 to 126
Characters v. Strings

• char values are always written in single quotes
• string values are always written in double quotes
• Examples:
  • char x = ‘H’;
  • string s = “Hello”, s1 = “Queens College”, s2 = “”, s3 = “A”;
  • char cs[] = “Hello”;
Strings and Arrays

• With a c-string we can change chars using array notation
  • char cs[] = “hello”;
  • cs[0] = ‘H’; // cs now stores Hello

• Same array notation works on string data
  • string s = “hello”;
  • s[2] = ‘L’; // s now stores heLlo
Example 2

• Write function to convert text to all lower case
• Title line: char toLower(char c)

```cpp
int main() {
    string s = "HELLO";
    for(int i = 0; i < 5; i++){
        s[i] = toLower(s[i]);
    }
    cout << s << endl;
    return 0;
}
```
String Class

• String is a class type
• Class types have methods, which are special functions to call on
• Use methods: `VariableName.MethodName(Arguments)`
  • The “.” is called the method selection operator
• Creating user-defined classes and methods is a CS 211 topic
String Methods

• String class methods include:
  • length(), size()
  • find(target), find(target, start_index)
  • rfind(target)
  • erase(index, amount)
  • replace(index, amount, addition)
  • insert(index, addition)
  • substr(index), substr(index, amount)
  • c_str()
Call on class methods

• Variable.Method(Required_Input)

• Example:

  string s = "Hello";
  cout << s.length() << endl;
  cout << s.size() << endl;
Example 3

• Rewrite function to convert any size string to all lower case
• Title line:  void toLower(string &a)

int main() {
    string s = “HELLO”;  
    toLower(s); 
    cout << s << endl;  
    return 0;  
}
Example 4

- Rewrite function to convert any size string to all lower case
- Title line: string toLower(string a)

```cpp
int main() {
    string s = "HELLO";
    string s1 = toLower(s);
    cout << s1 << endl;
    return 0;
}
```
c_str()

• There are two ways to store sequences of characters in C++
  • string – class type
  • char[] – c-string pointer (left over from C language)
• Work with string and if you absolutely need a c-string convert the string to it using .c_str()
• c-strings do not have methods, instead there are library functions
• To use c-string functions #include<cstring>
• These function names begin with str
c-string functions

char cs[] = “Queens College”;
cout << strlen(cs) << endl;
string s = “Hello”;
cout << s.c_str() << endl;
cout << strlen(s.c_str()) << endl;
strcpy(cs, s.c_str());
cout << cs << endl;