NOI 2020 C++ Basics Course Content

Note 7

Strings

The generic types of C++ don’t allow working with strings (words, sentences, or any sequence of characters). For this C++ offers string objects.

A string object can hold strings of different lengths.

Declaring and initializing a string object

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

int main() {
  string str; //declaration
  str = “initial string”; //initialization
  string str1 = “Hello world”; // declaration and initialization
  return 0;
}
```

Reading a string

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

int main() {
  string str;
  cout<<”Enter a string:”;
  cin>>str;
  cout<<str;
  return 0;
}
Input: Hello World
Output: Hello

Note: This will only read a single word. The user entered “Hello
World” but cin will only read until the first space. That is cin stops
reading after the space after “Hello”.
```

Reading a full line

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

int main() {
  string str;
  cout<<”Enter a string:”;
  getline(cin,str);
  cout<<str<<endl;
  return 0;
}
Input: Hello World
Output: Hello World
```

Note: Here the getline function will read a complete line of input without stopping at the space.
getline(<input stream>, <variable name>)
If you are reading user input from the console use “cin” as the input stream.

String Operations

- Concatenation

  To add two strings ‘+’ operator can be used. This is called concatenation.

  ```
  #include <iostream>

  using namespace std;

  int main() {
    string str = “String 1”;
    string str1 = “String 2”;
    string newStr = str + str1;
    cout<<newStr<<endl;
    return 0;
  ```

```
}
Output: String 1String 2
```

- Append

Another way to add a string at the end of another string is to use the append function.

```
#include <iostream>

using namespace std;

int main() {
  string str = “String 1”;
  string str1 = “String 2”;
  string newStr = str.append(str1);
  cout<<newStr<<endl;
  return 0;
}
Output: String 1String 2
```

Note: The append function is much faster than the + operator. Therefore it is recommended to use the append function where possible.

- Length

The length function of a string object can be used to get the length of the string it’s storing.

```
#include <iostream>

using namespace std;

int main() {
  string str = “String 1”;
  int strlength = str.length();
  cout<<strlength <<endl;
  return 0;
}
Output: 8
```

Note: The length function and the size function are the same. Therefore you can use str.size() instead of str.length() as you wish.

- Access characters of the string

  Characters of a string object can be accessed and modified the same way of accessing an array.

  - Reading a character from a string

    ```
    #include <iostream>

    using namespace std;

    int main() {
      string str = “String 1”;
      char thirdCharacter = str[2];
      cout<<thirdCharacter<<endl;
      return 0;
    }
    Output: r
    ```

  - Changing a character of a string

    ```
    #include <iostream>

    using namespace std;

    int main() {
       string str = “String 1”;
       str[2] = ‘A’;
       cout<<str<<endl;
       return 0;
    }
    Output: StAing 1
    ```

Extra resources:

- Strings
  Video:
  - https://www.youtube.com/watch?v=-t07hCkdHjI
  - https://www.youtube.com/watch?v=5ZX6-ZzTZ98
  - https://www.youtube.com/watch?v=mp7Um7k-G4I
  - https://www.youtube.com/watch?v=gWWslxWx5QQ

  Reading:
  - http://www.cplusplus.com/reference/string/string/
  - https://www.w3schools.com/cpp/cpp_strings.asp
  - https://www.w3schools.com/cpp/cpp_strings_concat.asp
  - https://www.w3schools.com/cpp/cpp_strings_length.asp
  - https://www.w3schools.com/cpp/cpp_strings_access.asp