Strings in C++

Dr. Ferdin Joe John Joseph
Kamnoetvidya Science Academy

Using Strings in C++ Programs

- **String library <string> or <cstring>** provides functions to:

  - manipulate strings

  - compare strings

  - search strings

- **ASCII character code**

  - Strings are compared using their character codes

  - Easy to make comparisons (greater than, less than, equal to)

Using Strings in C++ Programs .. Cont.

- **Character Constant**

  - Integer value of a character
  - Represented with single quotes
  - ‘z’ is the integer value of z, which is 122

- **String in C++**

  - Series of characters treated as one unit
  - can include letters, digits, special characters +, -, *, …
  - String literal (string constants) enclosed in double quotes, for example: **“C++ course”**

Using Strings in C++ Programs .. Cont.

Example:

Write a C++ program that reads two initials and the last name of a person and displays a personalized message to the program user.

Analysis stage:

- **Input:**

    2 characters for the initials (e.g. first and second)

    1 string for the last name (e.g. last)

-**Output:**

    a message to welcome the user

Using Strings in C++ Programs .. Cont.

```
//A program to display a user’s name with a welcome message
#include <iostream>
#include <string>
using namespace std;
int main ( )
{  char first, second;       //input and output: first and second initials
   string last;              //input and output: last name
  // Enter letters and print message.
   cout<<"Enter 2 initials for your first and second names and last name: " ;
   cin >> first >> second >> last;
   cout<< "Hello "<<first<< ". " << second<<". " <<last<< endl;
}
```

Using build in library.

```
#include <iostream>
#include <string>
using namespace std;
int main ()
{string str1 = "Hello";
   string str2 = "World";
   string str3;
   int  len ;
  str3 = str1;            // copy str1 into str3
  cout << "str3 : " << str3 << endl;
 // concatenates str1 and str2
  str3 = str1 + str2;
  cout << "str3 : " << str3 << endl;
  len = str3.size();
  cout << "str3.size() :  " << len <<endl;
 }
```

Fundamentals of Strings in C++

- String can be array of characters ends with null character '\0'.

char color [ ] = "green" ;

-- this creates 6 element char array, **color**, (last element is '\0')

g	**r**	**e**	**e**	**n**	**\0**

-- color can be declared also as :

char color [ ] = {'g', 'r', 'e', 'e', 'n', '\0' };

char color [6] = {'g', 'r', 'e', 'e', 'n', '\0' };

Fundamentals of Strings in C++ .. Cont.

- String can be constant pointer that points to the string’s first character.

Example:

char *colorPtr = “green” ;

-- this creates pointer variable **colorPtr** that points to the string “**green**” that is stored somewhere in memory

g	**r**	**e**	**e**	**n**	**\0**

-- value of variable **colorPtr** is the address of its first character(g)

Example

```
int main( ) {
        char firstName[] = “Ferdin";
        char *lastName  = “Joe";

        cout<<"First Name: "<<firstName<<endl;
        cout<<"Last Name: "<<lastName<<endl;
        int i=0;
        cout<<"FirstName: ";
        while (firstName[i] != '\0')
                cout<<firstName[i++];
        i=0;
        cout<<"\nLast Name: ";
        while (lastName[i] != '\0')
                cout<<lastName[i++]; }
```

Fundamentals of Strings in C++ .. Cont.

- **Reading Strings**

  - Assign input to character array, for example

```
                    char word [ 20 ];
            cin >> word;
             cout<<word<<endl;
```

-- this reads characters until a space, tab, newline, or end-of-file is encountered.

-- the string should be less than 19 characters, the 20th is for the null character (‘\0’).

Problem: read characters until the first white space

Fundamentals of Strings in C++ .. Cont.

- solution: To read an entire line of text into an array, C++ uses: **getline** function as follows:

  cin.getline (array, array size, delimiter character);

  - **getline** will copy input into specified array until either

    -- one less than the size is reached

    -- the delimiter character is input

  - **Example:**

```
char word [20] ;
cin.getline ( word, 20, ‘\n’ );
```

String Manipulation Functions

Function	Description
char *strcpy(char *s1, const char *s2);	Copies string s2 into the character array s1. The value of s1 is returned.
char *strncpy(char *s1, const char *s2, size_t n);	Copies at most n characters of string s2 into the array s1. The value of s1 is returned.
char *strcat (char *s1, const char *s2);	Appends string s2 to string s1. The value of s1 is returned.
char *strncat (char *s1, const char *s2, size_t n);	Appends at most n characters of string s2 to string s1. The value of s1 is returned.

String Manipulation Functions .. Cont.

int strcmp(const char *s1, const char *s2);	Compares string s1 with string s2. The function returns a value of zero, less than zero or greater than zero if s1 is equal to, less than or greater than s2, respectively.
int strncmp(const char *s1, const char *s2, size_t n);	Compares up to n characters of string s1 with string s2. The function returns zero, less than zero or greater than zero if s1 is equal to, less than or greater than s2, respectively.

String Manipulation Functions .. Cont.

Size_t strlen(const char *s);	Determines the length of string s. The number of characters preceding the terminating null character is returned.

String Manipulation Functions .. Cont.

1- strcpy(s1, s2) → s1 = s2   Copies string s2 into string s1.

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
            char str1[] =“Ferdin";
            char *str2 = “Joe";

            strcpy(str1,str2);

            cout<<str1<<endl;
 }
```

2- strncpy(s1, s2) → s1[n] = s2[n]

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

int main() {
            char str1[] =  ”**********";
            char *str2 = “$$$$$$$$$$";

            strncpy(str1,str2,5);

            cout<<str1<<endl;
}
```

3- strcat(s1, s2) → s1 = s1+s2

Concatenates string s2 onto the end of string s1.

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

int main() {
        char str1[24] =  ”Kamnoetvidya ";
        char *str2 = “Science Academy";

        strcat(str1,str2);

        cout<<str1<<endl;
 }
```

4- strncat(s1, s2,n) → s1 = s1+s2[n]

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

int main() {
        char str1[24] =  ”Rayong";
        char *str2 = “Kamnoetvidya Science Academy";

        strncat(str1,str2,10);

        cout<<str1<<endl;
}
```

5- strcmp(s1, s2) → 0 if s1 = s2

→ -1 if s1 < s2

→ 1 if s1 > s2

Symbols < … < numbers < … < capital letters < …. < smal letters.

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
        char str1[20] ;
        char str2[20] ;
        cin.getline(str1,20);
        cin.getline(str2,20);

        if (strcmp(str1,str2))
                if (strcmp(str1,str2) == 1)
                        cout<<str1<<" > "<<str2<<endl;
                else
                        cout<<str1<<" < "<<str2<<endl;
        else
                cout<<str1<<" = "<<str2<<endl;  }
```

6- strncmp(s1, s2,n)

→ 0 if s1[n] = s2[n]

→ -1 if s1[n] < s2[n]

→ 1 if s1[n] > s2[n]

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
          char str1[20] ;
          char str2[20] ;
          cin.getline(str1,20);
          cin.getline(str2,20);

          if (strncmp(str1,str2,1))
                    if (strcmp(str1,str2,1) == 1)
                              cout<<str1<<" > "<<str2<<endl;
                    else
                              cout<<str1<<" < "<<str2<<endl;
          else
                    cout<<str1<<" = "<<str2<<endl;  }
```

7- strlen(s) → How many characters in s

is a function that accepts a string, defined as an array of characters, and returns the number of characters in the string excluding null character

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
            char s1[] = “KVIS School";
            char *s2 = “Kamnoetvidya Science Academy";
            cout<<s1<<" Consists of "<<strlen(s1)<<" Characters.\n";
            cout<<s2<<" Consists of "<<strlen(s2)<<" Characters.\n";
}
```