11/04/19

string s1 = “hello”;

string s2 = “help”;

string s3 = “helping”;

string s4 = “hElp”

if (s1 < s2) // true (‘p’ > ‘l’)

…

if (s2 < s3) //true (‘ing’ > ‘’)

…

if (s2 < s4) //false if ASCII (‘e’ > ‘E’), true for EBCDIC (‘e’ < ‘E’)

…

C strings

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 5 | ‘H’ | ‘e’ | ‘l’ | ‘l’ | ‘o’ | ‘\0’ |

Treat the first character as number -> problem (limited size of string)

* two bytes? – more?

A marker of the end (a character -> ignore, null – numeric = 0)

‘\n’ – new line

‘\t’ – tab

‘\\’ - \

‘\’’ – ‘

‘\”’ – “

Escape characters

‘\0’ – zero byte (null byte in character)

char t[10] = {‘H’, ‘e’, ‘l’, ‘l’, ‘o’, ‘\0’) -> non itiliazed -> netural/ falsevecpt

“Hello” – zero byte -> implied

char t[10] = “Hello”;

char s[100]; - uninitialized

char s[100] = ‘’; or char s[100] = ‘\0’;

for (int k = 0; t[k] != ‘\0’; l++)

{

cout << t[k] << endl;

}

Ex:

char u[5] = “Hello”; -> failed to compile (too many initializer)

char v[5] = {‘H’, ‘e’, ‘l’, ‘l’, ‘o’}; //v(“Hello”)

* undefined behavior (no zero byte) ~ code assumes zero byte at the end

char v[5] = {}; -> all be zero bytes

Back to example:

char t[10] = “Hello”;

for (int k = 0; t[k] != ‘\0’; l++)

{

cout << t[k] << endl;

}

cout << t;

cin.getline(s, 100) – parameter (name of the array, limit to the input)

Limit 100 -> 99 interesting ints

Syntaxt: C++ - getline(Cin, s)

Cannot assign array or string to C==/C

COUT << SROOOPCOPY

Sstrcpy (s, t) // strcpy t(destination) and spirce fo;e\_

* Count 0 bytes -> continue (enver wending\_

!!!

No zero byte off

char t[10] = “Hello”;

cout << t;

cin.getline (s, 100);

cout << strlen(t); //writes 5, the length of the C string Hello

strcat[s, “1”]

* strcat, ‘H’; -> same

Additional functions to check the length

#include <cstring> //for strlen, strcpy -> strcpy – depreciated by Microsoft

* Warning (depreciated) -> regular warning

#define \_CRT\_SECURE\_NO\_WARNING -> before #include <cstring> using namespace std;

C string

if (s == t)

if (s < t)

WILL COMPILE!!! BUT ERROR

New Example:

strcpy(t, “Goodbyes”);

strcpy(s, “Goodness);

if (strcmp(t, s) < 0) //C++ strings: if (t < s) -> string compare

->

strcmp (a, b) //return int: negative if a comes before b, 0 if a==b, positive if a comes after b

C++ strings: a == b

C wtrings: strcmp (a, b) == 0

MOST COMMON MISTAKES IN C STRING

* Is a equal to b?
  + Wrong: if (strcmp(a, b)) //actually yields the opposite result -> 0 = false
  + Right: if (strcmp(a, b) == 0)
* Does a come before b?
  + Wrong: if (a < b) //will compile for C strings but not do what you want
  + Right: if(strcmp(a,b) > 0)

2-D Array

const int NWEEKS = 5;

const int NDAYS = 7;

int attendance [NWEEKS][NDAYS]; //attendance with 5 rows, 7 columns

…

cout << attendance [2][5] //row 2, column 5

for (int w = 0; w <NWEEKS; w++)

{

int t = 0;

for (int d = 0, d <NDAYS; d++)

t += attendance [w][d];

cout << “The total for week “ << w << “ is “ << t << endl;

}

const string dayNames [NDAYS} = {“Monday”, “Tuesday”, …, “Sunday”};;

//let weekends be 5, 6, 7

int grandTotal = 0;

for (int d = 4; d < NDAYS; d++)

{

int t = 0;

for (int w = 0; w < NWEEKS; w++)

t += attendance [w][d];

cout << “The total for “ << dayNames[d] << “ is “ << t << endl;

grandTotal += t;

}

cout << “Over the course of” << NWEEKS << “ weeks, weekend attendance was “ << grandTotal << endl;