Минестерство образования и науки Российской Федерации

ФГБОУ ВО Ульяновский государственный технический университет

Кафедра «вычислительная техника»

Лабораторная работа №13

«Строки и модули»

Выполнил студент

Группы ИВТАСбд-11

Долгов А. П.

Ульяновск, 2021

1. **Постановка задачи**

Требуется написать модуль, реализующий следующие функции, аналогичные части функций из библиотеки string:

* strlen
* strcpy
* strcat
* strcmp
* strchr
* strstr

Кроме этого, требуется написать программу, осуществляющую тест реализованных функций. Требуется ввести строки, являющиеся входными данными. Выполнить действие функции над ними. Вывести на экран результат выполнения функции

1. **Особенности реализации**

В файле gamma.h хранятся определения всех функций (они имеют такие же названия, как и функции из string, но с префиксом g). В файле «gamma.cpp» реализуются все определения функций:

size\_t \_\_cdecl gstrlen(const char\* \_Str)

{

int \_Count = 0;

while (\_Str[\_Count] != '\0')

\_Count++;

return \_Count;

}

char\* gstrcpy(char\* \_Destination, const char\* \_Source)

{

int id = 0;

while (\_Source[id] != '\0')

{

\_Destination[id] = \_Source[id];

id++;

}

\_Destination[id] = \_Source[id];

return \_Destination;

}

char\* gstrcat(char\* \_Destination, const char\* \_Source)

{

int dID = 0;

int sID = 0;

while (\_Destination[dID] != '\0')

dID++;

while (\_Source[sID] != '\0')

{

\_Destination[dID] = \_Source[sID];

dID++;

sID++;

}

\_Destination[dID] = \_Source[sID];

return \_Destination;

}

int gstrcmp(char const\* \_Str1, char const\* \_Str2)

{

int i = 0;

while (\_Str1[i] != '\0' && \_Str2[i] != '\0' && \_Str1[i] == \_Str2[i]) i++;

int k = \_Str1[i] - \_Str2[i];

if (k == 0) return 0;

if (k < 0) return -1;

if (k > 0) return 1;

}

char\* gstrchr(const char\* \_String, int \_Ch)

{

int id = 0;

while (\_String[id] != '\0')

{

if (\_String[id] == \_Ch)

return &(((char\*)\_String)[id]);

id++;

}

return NULL;

}

char\* gstrstr(char\* \_String, const char\* \_SubString)

{

int id = 0;

int id2 = 0;

int countStr2 = strlen(\_SubString);

char\* ptr = nullptr;

while (\_String[id] != '\0' && \_SubString[id2] != '\0')

{

if (\_SubString[id2] != \_String[id])

id2 = 0;

if (id2 == 0)

ptr = &(\_String[id]);

if (\_SubString[id2] == \_String[id])

id2++;

id++;

}

if (countStr2 == id2)

return ptr;

else

return nullptr;

}

const char\* gstrstr(const char\* \_String, const char\* \_SubString)

{

int id = 0;

int id2 = 0;

int countStr2 = strlen(\_SubString);

const char\* ptr = nullptr;

while (\_String[id] != '\0' && \_SubString[id2] != '\0')

{

if (\_SubString[id2] != \_String[id])

id2 = 0;

if (id2 == 0)

ptr = &(\_String[id]);

if (\_SubString[id2] == \_String[id])

id2++;

id++;

}

if (countStr2 == id2)

return ptr;

else

return nullptr;

}

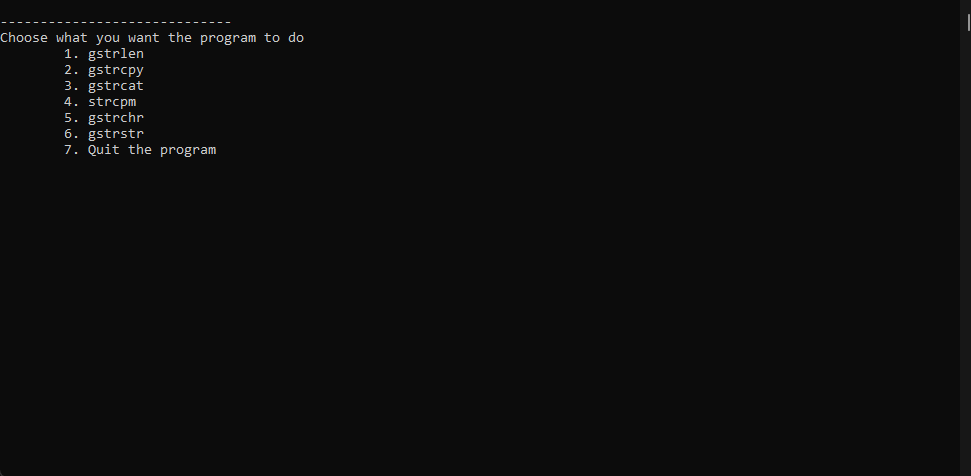
Строка является обычным массивом символов char. При этом в каждой строке должен находится так называемый нуль оператор (имеет код 0, в C можно записать так: '\0'), который служит для обозначения конца строки. Без него функции в данной работе, также как и в самой библиотеке string.h, работать не будут.

gstrlen позволяет определить длину строки. gstrcpy копирует одну строку в начало другой. gstrcat объединяет две строки в одну. gstrcmp сравнивает две строки. gstrchr указывает первое вхождения введённого символа в введённой строке. gstrstr указывает первое вхождения одной введённой строки в другую.

А в файле «source.cpp» написан код проверки всех функций (с помощью меню).

1. **Тестирование**

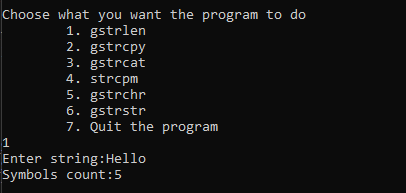
При запуске программы появляется меню выбора функции:

****

Поочерёдно проверим работу каждой функции:

*gstrlen.*

Первым делом предлагается ввести строку. После её ввода высвечивается количество символов в строке:



*gstrcpy.*

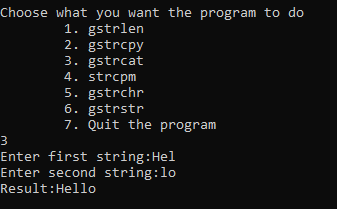
Предлагается ввести две строки, после чего выводится первая строка со скопированной в неё второй строкой:



Однако из-за того, что копируется в том числе и символ окончания строки ‘\0’, выводится лишь то, что находится до этого символа, т. е. вторая строка (в данном случае: «How are you?»).

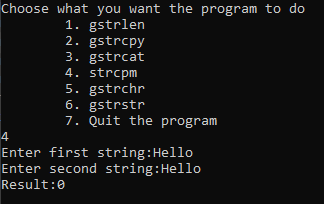
*gstrcat.*

Предлагается ввести две строки. Затем выводится эти же строки, но объединённые в одну:

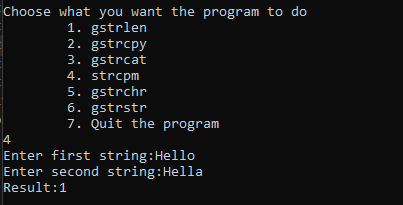


*gstrcmp.*

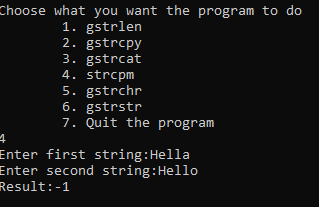
Предлагается ввести две строки, которые будут сравниваться. Если строки одинаковые, то будет выведен 0:



Если же разные, и при этом последний символ первой строки имеет кодовое значение больше чем последний символ второй строки, то будет выведено 1:

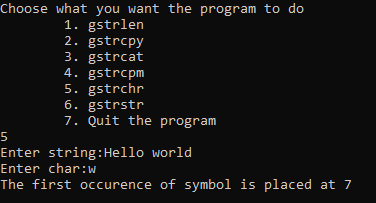
****

Если же наоборот, последний символ первой строки имеет кодовое значение меньшее чем последний символ второй строки, то будет выведено 2:



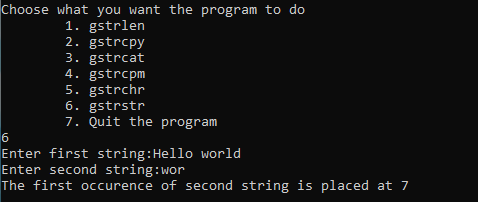
*gstrchr.*

Предлагается ввести строку и символ. После чего выводится порядковый номер первого вхождения введённой буквы в ведённой строке (слева на право, начиная с единицы):

****

gstrstr.

Предлагается ввести две строки. Затем выводится порядковый номер символа первого вхождения второй строки в первую (слева на право, начиная с единицы):



1. **Выводы о проделанной работе**

Программа протестирована и работает успешно. В ней реализованы часть функции из библиотеки string.h, работу которых можно проверить при запуске программы с помощью специального меню.