Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Алгоритмізація та програмування

Базові концепції програмування

ЗВІТ ДО

ЛАБОРАТОРНОЇ РОБОТИ №6

Варіант №17

Виконав: студент 1-го курсу

гр. ТР-12

Каркушевський В.Л. (П.І.Б.)

Оцінка \_\_\_\_\_\_\_\_\_ Перевірила: Смаковська Г.М

Дата «12»Грудня 2021 \_\_\_\_\_\_\_\_\_\_\_(П.І.Б., підпис)

Київ – 2021

**ЛР №6**

Разработать функцию, которая выполняет ту обработку символьной строки, которая определена в Вашем индивидуальном задании. При реализации функции запрещается пользоваться функциями библиотек языка C.

     Примечания:

1. В большинстве заданий необходимо предусмотреть работу функции при некорректном задании ее параметров.
2. В тех заданиях, где применен термин "слово", под ним понимается любая последовательность символов, ограниченная любым числом пробелов и/или началом/концом строки.

**Варианты индивидуальных заданий**

**ТР-12**

**Вариант 17**

Функция переписывает все символы строки в обратном порядке.

**Теоритичні відомості**

Рядок – це послідовність ASCII або UNICODE символів.

Рядки в C, як і більшості мов програмування високого рівня розглядаються як окремий тип, що входить до системи базових типів мови. Так як мова C за своїм походженням є мовою системного програмування, то рядковий тип даних C як відсутній, а як рядків С використовуються звичайні масиви символів.

Історично склалося два уявлення формату рядків:

формат ANSI;

рядки з завершальним нулем (використовується в C).

Рядки реалізуються за допомогою масивів символів. Тому оголошення ASCII рядка має наступний синтаксис:

char ім'я[довжина];

Оголошення рядка С має той же синтаксис, що і оголошення одномірного символьного масиву. Довжина рядка повинна бути ціле значення (у стандарті C89 – константа, у стандарті C99 може бути виразом). Довжина рядка вказується з урахуванням одного символу на зберігання завершального нуля, тому максимальна кількість значущих символів у рядку на одиницю менша за її довжину. Наприклад, рядок може містити максимально двадцять символів, якщо оголошено так:

char str [21];

Ініціалізація рядка С здійснюється при її оголошенні, використовуючи наступний синтаксис:

char str[довжина] = рядковий літерал;

Рядковий літерал – рядок ASCII символів укладених у подвійні лапки. Приклади оголошення рядків з ініціалізацією:

char str1[20] = "Введіть значення: ", str2[20] = "";

**Використання рядків**

Рядки корисно використовувати тоді, коли вам необхідно виконувати різні операції з текстовою інформацією. Наприклад, якщо ви бажаєте, щоб користувач вводив ім'я в програму, ви повинні використовувати рядок. Використання функції scanf() для введення рядка працює, але це може призвести до переповнення буфера. Адже вхідний рядок може виявитися більшим, ніж розмір рядка-буфера. Є кілька способів для вирішення цієї проблеми, але найпростіший спосіб це використовувати fgets() функцію, яка оголошена в заголовному файлі <stdio.h>.

Коли fgets() зчитує вхідні дані від користувача, вона читатиме всі символи, крім останнього. Після цього на кінець ліченого рядка, fgets() помістить нульовий термінатор. Функція fgets() зчитуватиме символи доти, доки користувач не натисне Enter. Давайте подивимося приклад використання fgets():

Функція fgets() зчитує до num-1 символів з файлу stream і поміщає в масив символів, який вказує str. Символи зчитуються доти, доки не зустрінеться символ «новий рядок», EOF або до досягнення вказаної межі. Після закінчення зчитування масив str відразу після останнього ліченого символу поміщається нульовий символ. Символ нового рядка при зчитуванні буде збережений і стане частиною масиву str.

У разі успіху fgets() повертає str, при невдачі повертається NULL. У разі помилки читання вміст масиву, на який вказує str, не визначено. Оскільки як у разі помилки, так і при досягненні кінця файлу повертається null, для визначення того, що саме сталося необхідно використовувати feof() або ferror

Символ \0 вважається символом NULL . Він використовується для позначення кінця рядка C.

C рядок-це покажчик, що вказує на масив символів з \0 в кінці. Таким чином, наступне буде допустимим представленням рядків C.

**Опис завдання**

Імпортуємо бібліотеку #include <stdio.h> для таких команд як printf, fgets і тд.

В методі main оголошуємо масив типу char в який будемо розміщувати наш рядок.

За допомогою команди fgets вводимо рядок.

Далі за допомогою циклу for проходимо всі елементи і шукаєте елемент '\0' тобто кінець рядка.

Присвоюємо змінній int size значення останнього елемента, тобто int size буде розміром нашого рядка.

Якщо розмір масиву буде парним, то за допомогою циклу for ми робимо заміну першого елементу на останній , другого на передостанній, і тд., аж поки не зміняться всі елементи. Якщо розмір масиву буде непарним, то ми так само за допомогою циклу for змінюємо всі елементи крім середнього.

В результаті виводимо на екран відсортований рядок за допомогою команди printf.

**Програмний код:**

#include <stdio.h>

int main(void) {

char myString[100];

printf( "Введіть рядок: " );

fgets( myString, 100, stdin );

printf( "\nВи ввели рядок : %s", myString );

int size = 0 ;

for (int i = 0; myString [i] != '\0'; i++){

size=i;

}

if(size%2==0){

int k= size ;

char temp ;

for(int i = 0 ; i < size/2 ; i++ ){

temp = myString[i];

myString[i] = myString[k];

myString[k] = temp;

k--;

}

}else{

int z= size-1 ;

char temp ;

for(int i = 0 ; i < (size-1)/2 ; i++ ){

temp = myString[i];

myString[i] = myString[z];

myString[z] = temp;

z--;

}

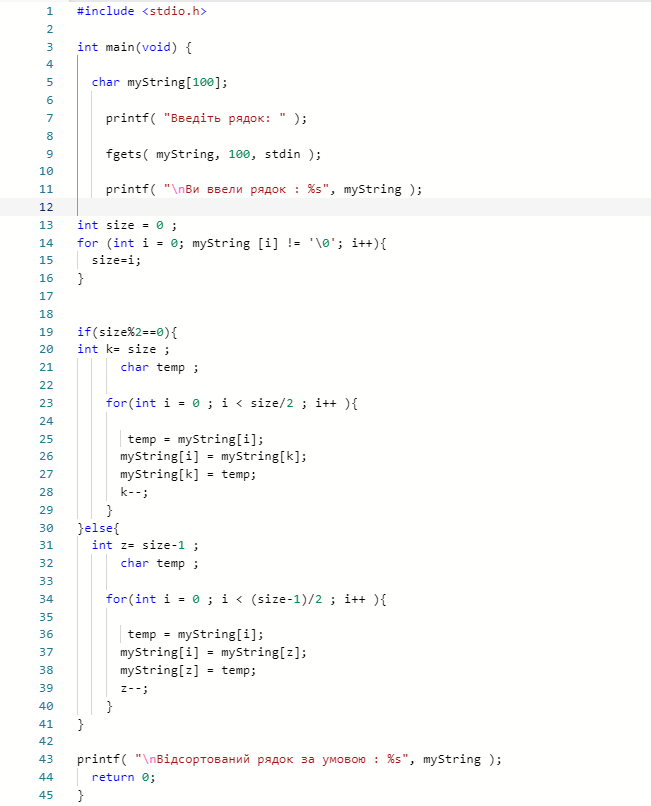
}

printf( "\nВідсортований рядок за умовою : %s", myString );

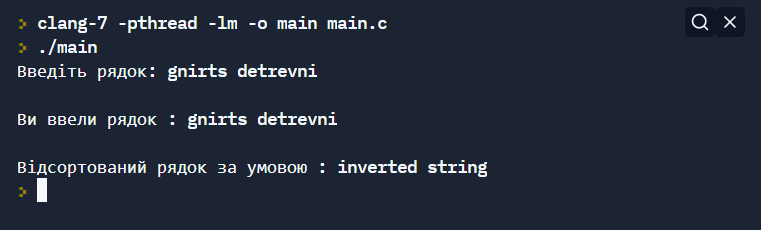
return 0;

}

**Код програми:**

****

**Результати рішення програми :**



**Висновок:**

На цій лабораторній роботі ми попрацювали з рядками, розробили функцію яка переписує всі символи в оберненому порядку без використання бібліотечних функцій мови С.