**Assignment - 18 A Job Ready Bootcamp in C++, DSA and IOT MySirG**

**String and Functions in C Language**

1. Write a function to calculate length of the string.

#include <stdio.h>

int string\_length(char str[])

{

int i;

for (i = 0; str[i] != '\0'; i++);

return i;

}

int main() {

char str[220];

printf("enter your string: ");

gets(str);

int length=string\_length(str);

printf("Length of the string is %d\n", length);

return 0;

}

2. Write a function to reverse a string.

#include <stdio.h>

#include <stdio.h>

int reverse\_string(char str[])

{

int i, len, temp;

len = strlen(str) - 1;

for (i = 0; i < len; i++, len--) {

temp = str[i];

str[i] = str[len];

str[len] = temp;

}

str[strlen(str)] = '\0';

return str;

}

int main() {

char myString[220];

printf("enter your string: ");

fgets(myString, 220, stdin);

int reverse=reverse\_string(myString);

printf("Reverse string is %s", reverse);

return 0;

}

3. Write a function to compare two strings.

#include <stdio.h>

#include <string.h>

int compare\_strings(char str1[], char str2[])

{

int len1 = strlen(str1);

int len2 = strlen(str2);

if (len1 != len2) {

return 0;

}

int i;

for (i = 0; i < len1; i++) {

if (str1[i] != str2[i]) {

return 0;

}

}

return 1;

}

int main()

{

char myString1[] = "Hello";

char myString2[] = "World";

if (compare\_strings(myString1, myString2)) {

printf("Strings are equal.\n");

} else {

printf("Strings are not equal.\n");

}

return 0;

}

4. Write a function to transform string into uppercase.

#include <stdio.h>

#include <string.h>

int transform\_string(char str[])

{

int i;

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

if(str[i]>=97&&str[i]<=122)

str[i]=str[i]-32;

}

return i;

}

int main() {

char str[220];

printf("Enter the string in lowercase: ");

fgets(str, 220, stdin);

int uppercase=transform\_string(str);

printf("Uppercase string: %s", str);

return 0;

}

5. Write a function to transform a string into lowercase.

#include <stdio.h>

#include <string.h>

int transform\_lowercase(char str[])

{

int i;

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

if(str[i]>=65&&str[i]<=90 )

str[i]=str[i]+32;

}

return i;

}

int main() {

char str[220];

printf("Enter the string in uppercase: ");

fgets(str, 220, stdin);

int lowercase=transform\_lowercase(str);

printf("Lowercase string: %s", str);

return 0;

}

6. Write a function to check whether a given string is an alphanumeric string or not.

(Alphanumeric string must contain at least one alphabet and one digit)

#include <string.h>

#include <stdbool.h>

#include <ctype.h>

bool isAlphaNumeric(char str[]) {

bool has\_alpha = false;

bool has\_digit = false;

for (int i = 0; i < strlen(str); i++) {

if (isalpha(str[i])) {

has\_alpha = true;

} else if (isdigit(str[i])) {

has\_digit = true;

}

if (has\_alpha && has\_digit) {

return true;

}

}

return false;

}

int main() {

char str[220];

printf("Enter the string: ");

fgets(str, 220, stdin);

if (isAlphaNumeric(str))

{

printf("%s is alphanumeric.\n", str);

} else {

printf("%s is not alphanumeric.\n", str);

}

return 0;

}

7. Write a function to check whether a given string is palindrome or not.

#include <stdio.h>

#include <string.h>

#include <ctype.h>

int is\_palindrome(char str[]);

int main() {

char str[100];

printf("Enter a string: ");

fgets(str, 100, stdin);

str[strcspn(str, "\n")] = '\0';

if (is\_palindrome(str)) {

printf("%s is a palindrome\n", str);

} else {

printf("%s is not a palindrome\n", str);

}

return 0;

}

int is\_palindrome(char str[]) {

int i, j, len;

len = strlen(str);

j = len - 1;

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

while (!isalnum(str[i])) i++;

while (!isalnum(str[j])) j--;

if (tolower(str[i]) != tolower(str[j])) {

return 0;

}

j--;

}

return 1;

}

8. Write a function to count words in a given string.

#include <stdio.h>

#include <string.h>

int countWords(char str[]) {

int count = 0;

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

if (str[i] == ' ' && str[i+1] != ' ') {

count++;

} else if (str[i+1] == '\0' && str[i] != ' ') {

count++;

}

}

return count;

}

int main() {

char str[220];

printf("Enter your string: ");

fgets(str, 220, stdin);

int wordCount = countWords(str);

printf("Number of words: %d\n", wordCount);

return 0;

}

9. Write a function to reverse a string word wise. (For example if the given string is

“Mysirg Education Services” then the resulting string should be “Services Education

Mysirg” )

#include <stdio.h>

#include <string.h>

void swap(char str[], int i, int j) {

char temp;

while (i <= j) {

temp = str[i];

str[i] = str[j];

str[j] = temp;

i++;

j--;

}

}

int main() {

char str[200];

int i = 0, start = 0, end = 0, flag = 0;

printf("Enter your string: ");

gets(str);

while (str[i] != '\0') {

while (str[i] != ' ') {

if (str[i] == '\0') {

flag = 1;

break;

}

end++;

i++;

}

swap(str, start, end-1);

if (flag == 1)

break;

start = end + 1;

end = start;

i++;

}

swap(str, 0, i-1);

printf("%s", str);

return 0;

}

10. Write a function to find the repeated character in a given string.

#include <stdio.h>

#include <string.h>

void find\_repeated\_char(char str[]) {

int count[256] = {0};

int length = strlen(str);

int i;

printf("Repeated characters: ");

for (i = 0; i < length; i++) {

count[str[i]]++;

}

for (i = 0; i < length; i++) {

if (count[str[i]] > 1) {

printf("%c ", str[i]);

count[str[i]] = 0;

}

}

printf("\n");

}

int main() {

char str[100];

printf("Enter a string: ");

fgets(str, 100, stdin);

find\_repeated\_char(str);

return 0;

}