(Day:07\08\2025)

1. Write a program to find the length of a string without using strlen().

Input=User types a string.

Process=Loop through each character to count.

Output=the length of the string.

#include <stdio.h>

int main()

{

char str[100];

int i = 0;

printf("Enter a string: ");

fgets(str, sizeof(str), stdin);

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

{

i++;

}

if (i > 0 && str[i - 1] == '\n')

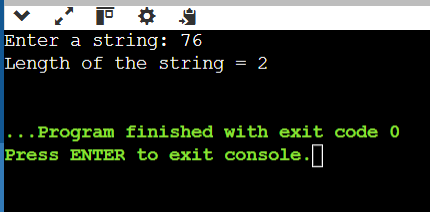
{

i--;

}

printf("Length of the string = %d\n", i);

}



1. Write a program to copy one string to another.

Input=User enters a string.

Process=Each character of the source string is copied one by one.

Output=The copied string is printed.

Program:

#include <stdio.h>

int main()

{

char source[100], destination[100];

int i = 0;

printf("Enter a string to copy: ");

fgets(source, sizeof(source), stdin);

while (source[i] != '\0')

{

if (source[i] == '\n')

{

source[i] = '\0';

break;

}

i++;

}

i = 0;

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

destination[i] = source[i];

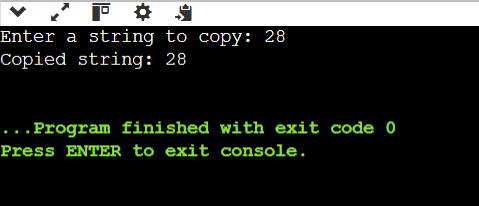
i++;

}

destination[i] = '\0';

printf("Copied string: %s\n", destination);

}



1. Write a program to concatenate two strings.

Input=Two strings from the user.

Process=Null-terminate the result.

Output=the concatenated string.

Program:

include <stdio.h>

int main()

{

char str1[100], str2[100];

int i = 0, j = 0;

printf("Enter the first string: ");

fgets(str1, sizeof(str1), stdin);

printf("Enter the second string: ");

fgets(str2, sizeof(str2), stdin);

while (str1[i] != '\0')

{

if (str1[i] == '\n')

{

str1[i] = '\0';

break;

}

i++;

}

while (str2[j] != '\0')

{

if (str2[j] == '\n')

{

str2[j] = '\0';

break;

}

j++;

}

i = 0;

while (str1[i] != '\0')

{

i++;

}

j = 0;

while (str2[j] != '\0')

{

str1[i] = str2[j];

i++;

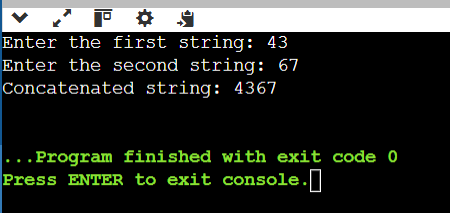
j++;

}

str1[i] = '\0';

printf("Concatenated string: %s\n", str1);

}



1. Write a program to compare two strings.

Input=Two strings from the user.

Process=Compare characters one by one from both strings.

Output=the strings are equal or not.

Program:

#include <stdio.h>

int main()

{

char str1[100], str2[100];

int i = 0, flag = 0;

printf("Enter the first string: ");

fgets(str1, sizeof(str1), stdin);

printf("Enter the second string: ");

fgets(str2, sizeof(str2), stdin);

while (str1[i] != '\0')

{

if (str1[i] == '\n')

{

str1[i] = '\0';

break;

}

i++;

}

i = 0;

while (str2[i] != '\0')

{

if (str2[i] == '\n')

{

str2[i] = '\0';

break;

}

i++;

}

/

i = 0;

while (str1[i] != '\0' || str2[i] != '\0')

{

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

{

flag = 1;

break;

}

i++;

}

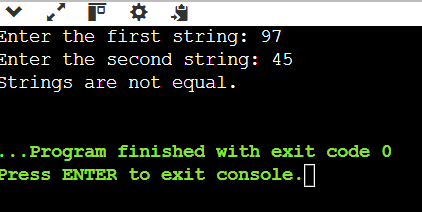
if (flag == 0)

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

else

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

}



1. Write a c program to count vowels and consonants in a string.

Input=User enters a string.

Process=it as a vowel or consonant.

Output=Number of vowels and consonants.

Program:

#include <stdio.h>

int main()

{

char str[100];

int i = 0, vowels = 0, consonants = 0;

printf("Enter a string: ");

fgets(str, sizeof(str), stdin);

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

{

char ch = str[i];

if ((ch >= 'A' && ch <= 'Z') || (ch >= 'a' && ch <= 'z'))

{

if (ch >= 'A' && ch <= 'Z')

{

ch = ch + 32;

}

if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')

{

vowels++;

}

else

{

consonants++;

}

}

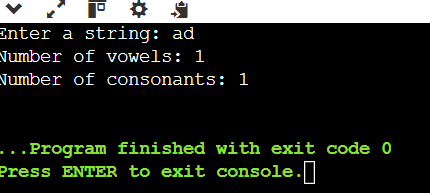
i++;

}

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

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

}



1. Write a program to convert lowercase to uppercase and vice versa.

Input=User enters a string.

Process=its uppercase, convert to lowercase.

Output=Print the converted string.

Program:

#include <stdio.h>

int main()

{

char str[100];

int i = 0;

printf("Enter a string: ");

fgets(str, sizeof(str), stdin);

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

{

if (str[i] >= 'a' && str[i] <= 'z')

{

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

}

else if (str[i] >= 'A' && str[i] <= 'Z')

{

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

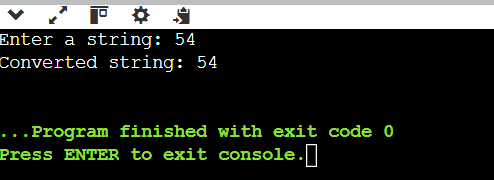
}

i++;

}

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

}



1. Write a program to check if a string is palindrome.

Input=User enters a string.

Process=it's not a palindrome.

Output=the string is a palindrome or not.

Program:

#include <stdio.h>

#include <string.h>

int main()

{

char str[100];

int i, length, isPalindrome = 1;

printf("Enter a string: ");

fgets(str, sizeof(str), stdin);

length = strlen(str);

if (str[length - 1] == '\n')

{

str[length - 1] = '\0';

length--;

}

for (i = 0; i < length / 2; i++)

{

if (str[i] != str[length - i - 1])

{

isPalindrome = 0;

break;

}

}

if (isPalindrome)

{

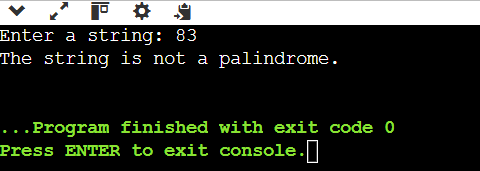
printf("The string is a palindrome.\n");

} else {

printf("The string is not a palindrome.\n");

}

}



8. Write a program to reverse a string.

Input=User enters a string.

Process=Swap characters from both ends.

Output=Print the reversed string.

Program:

#include <stdio.h>

#include <string.h>

int main()

{

char str[100], temp;

int i, j;

printf("Enter a string: ");

fgets(str, sizeof(str), stdin);

int length = strlen(str);

if (str[length - 1] == '\n')

{

str[length - 1] = '\0';

length--;

}

i = 0;

j = length - 1;

while (i < j)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

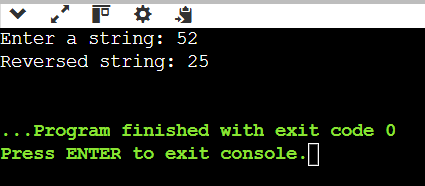
i++;

j--;

}

printf("Reversed string: %s\n", str);

}



9.Write a program to count words in a string.

Input=User types a string.

Process=Loop through the string.

Output=Print the total number of words.

Program:

#include <stdio.h>

#include <ctype.h>

int main()

{

char str[200];

int i = 0, wordCount = 0;

printf("Enter a string: ");

fgets(str, sizeof(str), stdin);

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

{

if (!isspace(str[i]) && (i == 0 || isspace(str[i - 1])))

{

wordCount++;

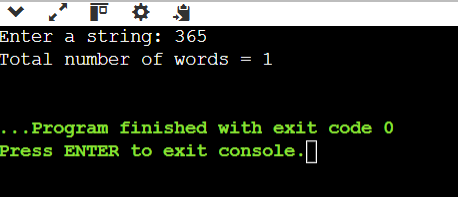
}

i++;

}

printf("Total number of words = %d\n", wordCount);

}



10. Write a program to find the frequency of each character in a string.

Input=A string entered by the user

Process=Count the frequency of each character.

Output=character and its corresponding frequency .

Program:

#include <stdio.h>

#include <string.h>

int main()

{

char str[1000];

int freq[MAX] = {0};

printf("Enter a string: ");

fgets(str, sizeof(str), stdin);

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

{

unsigned char ch = str[i];

if (ch != '\n' && ch != ' ')

{

freq[ch]++;

}

}

printf("\nCharacter Frequencies:\n");

for (int i = 0; i < MAX; i++)

{

if (freq[i] > 0)

{

printf("'%c' = %d\n", i, freq[i]);

}

}

}

