AIM:

A Program to find string length. <u>C program to find length of a string without using strlen function</u>, <u>recursion</u>.

String	Length
➢ " ➢ 'string' ➢ 'Fun with C' ➢ 'no strings attached'	➢ 0 ➢ 6 ➢ 10 ➢ 19

1)Length of string in C language.

Description:

C program to find length of a string, for example, the length of the string "C programming" is 13 (space character is counted). The null character isn't counted when calculating it. To find it, we can use strlen function of "string.h."

Program:

```c
#include <stdio.h>
#include <string.h>

int main()
{
  char a[100];
  int length;

 printf("Enter a string to calculate its length\n");
  gets(a);

 length = strlen(a);

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

 return 0;
}
```

Output of program:
Enter a string to calculate it's length
Programming simplified
Length of entered string is = 22

2) String length in C without strlen.

Description: You can also find string length without strlen function. We create our function to find it. We scan all the characters in the string if the character isn't a null character then increment the counter by one. Once the null character is found the counter equals the length of the string.

Program:

```c
#include <stdio.h>
int main()
{
  char s[1000];
  int c = 0;

 printf("Input a string\n");
  gets(s);

 while (s[c] != '\0')
    c++;

 printf("Length of the string: %d\n", c);

 return 0;
}
```

3) Function to find string length:

Program:

```c
int string_length(char s[]) {
  int c = 0;
  while (s[c] != '\0')
    c++;

  return c;
}
```

4) C program to find length of a string using recursion.

Program:

```c
#include <stdio.h>
int string_length(char*);

int main()
{
   char s[100];
 gets(s);
 printf("Length = %d\n", string_length(s));
 return 0;
}

int string_length(char *s) {
   if (*s == '\0') // Base condition
     return 0;
 return (1 + string_length(++s));
}
```

5) Function to find string length using pointers.

Program:

```c
int string_length(char *s) {
    int c = 0;

    while(*s[c] != '\0')
      c++;

    return c;
}
```