

strlen() function in c

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

int main()
{
    char ch[]={'h', 'e', 'l', 'l', 'o', '\0'};
    printf("Length of string is: %d", strlen(ch));
    return 0;
}
```

without strlen() finding the length of string:

```
#include<stdio.h>
int main()
{
    char ch[]={'h', 'e', 'l', 'l', 'o', '\0'};
    int i=0, len=0;
    while(ch[i]!='\0')
    {
        len++;
        i++;
    }
    printf("Length of string is: %d", len);
    return 0;
}
```

strupr() function in c

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

int main()
{
    char str[ ] = "hello cse";
    printf("%s\n", strupr (str));
    return 0;
}
```

without using strupr() function:

```
#include<stdio.h>
int main()
{
    char str[] = "Hello cse";
    int i=0;
    while(str[i]!='\0')
    {
        if(str[i]>='a' && str[i]<='z')
            str[i]-=32;    //difference between a and A is 32
            i++;
    }
    puts(str);
    return 0;
}
```

strlwr() function in C

```
#include<stdio.h>
#include <string.h>
int main()
{
    char str[] = "CompuTer ScienCe ";
    printf("Given string is: %s\n",str);
    strlwr(str);
    printf("\nString after converting to the lowercase is: %s",str);
    return 0;
}
```

without using strlwr() function:

```
#include<stdio.h>
int main()
{
    char str[] = "Hello cse";
    int i=0;
    while(str[i]!='\0')
    {
        if(str[i]>='a' && str[i]<='z')
            str[i]+=32;    //difference between a and A is 32
            i++;
    }
    puts(str);
    return 0;
}
```

strcat() / strncat() function in C

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

```
int main()
{
    char dest[25] = "This is an example";
    char src[50] = " of strcat";
    strncat(dest, src, 9);
    puts(dest);
    strcat(dest, src);
    puts(dest);
    return 0;
}
```

without using strcat() function:

```
#include <stdio.h>
int main()
{
    char dest[25] = "This is an example";
    char src[50] = " of strcat";
    int i,j=0;
    for(i=strlen(dest);i++)
    {
        dest[i]=src[j];
        if(src[j]=='\0')
            break;
    }
}
```

```

        j++;
    }
    printf("After concatenation: %s",dest);
    return 0;
}

```

HomeWork: Try strncat() function

strcpy() / strncpy() function in C

```

#include <stdio.h>
#include <string.h>
int main()
{
    char src[] = "hello cse";
    char destNew[20], dest[14];

    strcpy(destNew, src);
    strncpy(dest, src, 4);
    printf("strcpy = %s\n strncpy result = %s",destNew,dest);

    return 0;
}

```

without using strcpy() function:

```

#include <stdio.h>
int main()
{
    char src[] = "hello cse",dest[20];
    int i,j;

```

```

i=j=0;
while(src[i]!='\0')
{
    dest[i]=src[i];
    i++;
}
dest[i]='\0';
printf("After copy result = %s",dest);
return 0;
}

```

HomeWork: Try strncpy() function

strcmp() / strncmp() function in C

```

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

```

```

int main()
{
    char str1[20], str2[20];
    int val;
    gets(str1);
    gets(str2);
    val = strcmp(str1,str2);
    if( val == 0 )
        printf("String1 and String2 are same");
    else if( val > 0 )
        printf("String1 is greater than String2");
}

```

```

else if( val < 0 )
    printf("String1 is smaller than String2");
val = strncmp(str1,str2,4);
printf("%d is output",val);
return 0;
}

```

without using strcmp() function:

```

#include <stdio.h>
int main()
{
    char str1[20], str2[20];
    int val=0,i;
    gets(str1);
    gets(str2);
    if(strlen(str1)!=strlen(str2))
        val=1;
    else
    {
        i=0;
        while(str1[i]!='\0')
        {
            if(str1[i]!=str2[i])
            {
                val=1;
                break;
            }
            i++;
        }
    }
}

```

```

    }
    if( val == 0 )
        printf("String1 and String2 are same");
    else
        printf("String1 and String 2 are different");
    return 0;
}

```

HomeWork: Try strcmp() function

strstr() function in C

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

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

```
int main()
```

```
{
```

```
    char s1[] = "Examples of String";
```

```
    char s2[] = "of";
```

```
    char* p;
```

```
    p = strstr(s1, s2);
```

```
    if (p)
```

```
    {
```

```
        printf("String found\n");
```

```
        printf("First occurrence of string '%s' in '%s' is '%s'", s2,
```

```
s1, p);
```

```
    }
```



```
    else
        printf("String not found\n");

    return 0;
}
```

HomeWork: Try without using strstr() function

strrev() function in C

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

int main()
{
    char s1[] = "Examples of String";

    printf("Given string: %s",s1);
    strrev(s1);
    printf("After reverse function %s",s1);

    return 0;
}
```

without using strcmp() function:

```
#include <stdio.h>
int main()
{
    char s1[] = "Examples of String";
    int len=strlen(s1),i,j,temp;
    i=0;j=len-1;
    while(i!=len/2)
    {
        temp=s1[i];
        s1[i]=s1[j];
        s1[j]=temp;

        i++;
        j--;
    }
    printf("After reverse function %s",s1);
    return 0;
}
```