DAY #34.
Saturday, July 4, 2026 10:57 AM

String functions #2

strcat() → It concatenate two strings
strcmp() → It compares two strings

} Library function provided 'C' predefined function } string.h

Syntax:- strcat (str1, str2)

Example:-

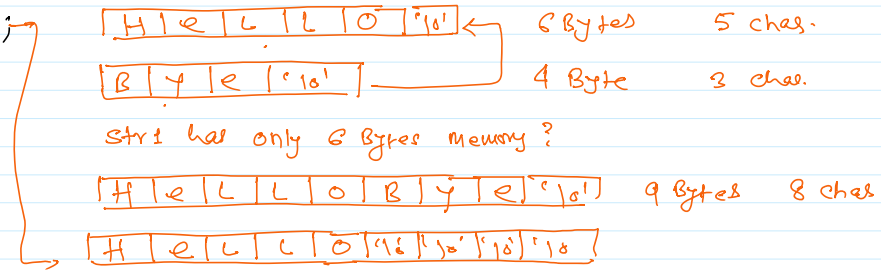
```
char str1[6] = "Hello";
char str2[4] = "Bye";
strcat (str1, str2);
printf("%s", str1);
```

↓

HelloBye

str1 = "Hello"; str2 = "Bye"

str1 = "HelloBye";



strncat() → You have to specify how many characters you want to concatenate?

strncat (str1, str2, n)

strncat (str1, str2, 2);

↓

HelloBy

strcmp()

Syntax: strcmp (str1, str2) returns 0 if str1 & str2 are equal
returns nonzero value.

```
char str1[] = "Hello";
char str2[] = "Bye";
int result;
  result = strcmp (str1, str2);

if (result == 0)
   printf("\n Strings are same";
else
   printf("\n Strings are not same";
```

- Write C program to compare two strings without using any library function (strcmp())

```
char str1[] = "Hello";
char str2[] = "Bye";

if (strlen(str1) != strlen (str2))
{   printf("\n strings are not same";
    return .
}
int flag = 0;
int len;
```

```
if (strlen (str1) < strlen (str2))
        len = strlen (str1)
else if ( strlen (2) < strlen (str1) )
        len = strlen (str2)
else
        len = strlen (str1);
```

len = 3

str1 = "Hello";
str2 = "Xyabc";

```
else
        len = strlen(str1);
for(int i=0; i<len; i++)
```

```
len = strlen(str1);
for(int i=0; i<len; i++)
{
    if(str1[i] != str2[i])
    {
        flag = 1;
        break;
    }
}
if(flag == 1)
    printf("In strings are not same");
else printf("In strings are same");
```

i=0 0<5 True H != x T flag=1
" Strings are not same

strncmp()

```
result = strncmp(str1, str2, 2);
```

str1 = "Hello"
str2 = "Hexyz";