

Logic Building Assignment : 36

Create separate visual Studio project for each problem statement separately.

1. Write a program which accepts 2 strings from user and concat N characters of second string after first string. Value of N should be accepted from user. (Implement strncat() function).

Note : If third parameter is greater than the size of second string then concat whole string after first string.

Input : **"Marvellous Infosystems"**
 "Logic Building"
 5

Output : **"Marvellous Infosystems Logic"**

```
void StrNCatX(char *src, char * dest, int iCnt)
{
    // Filter
    while(*str != _____)           // Traverse first string till end
    {
        // Logic
    }
    while((* dest != '\0') && (iCnt != 0))           // Copy contents of
destination in source
    {
        // Logic
        iCnt- -;
    }
    *dest = '\0';
}
```

```
int main()
{
    char arr[50] = "Marvellous Infosystems";
    char brr[30] = "Logic Building"

    StrNCatX(arr,brr,);

    prnitf("%s",arr);                // Marvellous Infosystems Logic

    return 0;
}
```

2. Write a program which 2 strings from user and check whether contents of two strings are equal or not. (Implement strcmp() function).

Input : **"Marvellous Infosystems"**
 "Marvellous Infosystems"

Output : **TRUE**

```
BOOL StrCmpX(char *src, char * dest)
{
    // Filter
    while((*str != _____) && (*dest != '/0'))
    {
        // Logic
    }

    if(_____ || _____)
    {
        return TRUE;
    }
}
```

```
}  
else  
{  
    return FALSE;  
}  
}  
  
int main()  
{  
    BOOL bret = TRUE;  
    char arr[50] = "Marvellous Infosystems";  
    char brr[30] = "Marvellous Infosystems"  
  
    bret = StrCmpX(arr,brr);  
  
    if(bret == TRUE)  
    {  
        printf("Both strings are equal");  
    }  
    else  
    {  
        printf("Both strings are not equal");  
    }  
    return 0;  
}
```

3. Write a program which 2 strings from user and check whether first N contents of two strings are equal or not. (Implement strcmp() function).

Note : If third parameter is greater than the size of second string then concat whole string after first string.

Input : **"Marvellous Infosystems"**
 "Marvellous Logic Building"
 10

Output : **TRUE**

```
BOOL StrNCmpX(char *src, char * dest ,int iCnt)
{
    // Filter
    while(_____)
    {
        // Logic
    }
    if(_____)
    {
        return TRUE;
    }
    else
    {
        return FALSE;
    }
}
```

```
int main()
{
    BOOL bret = TRUE;
    char arr[50] = "Marvellous Infosystems";
    char brr[30] = "Marvellous Logic Building"

    bret = StrNCmpX(arr,brr,10);

    if(bret == TRUE)
    {
        printf("Both strings are equal");
    }
    else
    {
        printf("Both strings are not equal");
    }
    return 0;
}
```

4. Accept string from user and reverse the contents of that string by toggling the case.

Input : "aCBdef"

Output : "FEDcbA"

```
void StrRevTogX(char *str)
{
    // Filter
    char temp;
    char *first,*last;
    first = str;
    last = str;
```

```
while(*last != '\0')
{
    last++;
}
last--;
while(first <= last)
{
    // Swapping logic with case toggle
}
}

int main()
{
    char arr[50] = "Marvellous";

    StrRevTogX(arr);

    printf("%s",arr);    // SUOLLEVRAM

    return 0;
}
```

5. Accept string from user and check whether the string is palindrome or not without considering its case.

Input : **"1abccBA1"**

Output : **TRUE**

```
void StrPallindrome(char *str)
{
    // Filter
    // Logic
}
```

```
}
```

```
int main()
```

```
{
```

```
    BOOL bret;
```

```
    char arr[20] = "abccBa";
```

```
    bret = StrPallindrome(arr);
```

```
    if(bret == TRUE)
```

```
    {
```

```
        printf("String is pallindrome\n");
```

```
    }
```

```
    else
```

```
    {
```

```
        printf("String is not pallindrome\n");
```

```
    }
```

```
    return 0;
```

```
}
```