

Assignment 10

```
//mystrcmp
```

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

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

```
int mystrcmp(char*,char*);
```

```
void main(){
```

```
    char str1[]="abcd";
```

```
    char str2[]="abCd";
```

```
    int res=mystrcmp(str1,str2);
```

```
    if(res==0){
```

```
        printf("same string :0");
```

```
    }
```

```
    else{
```

```
        if(res== -1)
```

```
        {
```

```
            printf("not same:1");
```

```
        }
```

```
        else
```

```
            printf("not same:-1");
```

```
    }
```

```
}
```

```
int mystrcmp(char* str1, char* str2){
```

```
    if(strlen(str1)==strlen(str2)){
```

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

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

```

        //check asscii value
        if(str1[i]>str2[i])
            return 1;
        else
            return -1;
    }

}

return 0;//for loop exits by its end value (all iterations)

}
else{
    return NULL;
}
}

//mystrstr
#include<string.h>
#include<stdio.h>
char* mystrstr(char*,char*);
void main() {
    /*
    char str[]="abdabeabxyz";
    char sub[]="axy";*/
    char str[]="prachiti";
    char sub[]="chiti";

    char* sub_str=mystrstr(str,sub);
    printf("%s",sub_str);
}

```

```

char* mystrstr(char* str,char* sub) {
    //check the length first
    if(strlen(str)>=strlen(sub)) {
        int p,len;
        //compare sub string with sub
        for(int i=0; sub[i]!='\0'; i++) {
            //i is for sub string
            //check each sub element with entire string
            for(int j=0; str[j]!='\0'; j++) {
                //j for string
                //compare
                if(sub[i]==str[j]) {
                    //here we get 1st match char .. now from j+1 check for
                    another element in sub

                    int x=i+1;

                    len=1;//starting from 1 bcz we got 1st elemrnt already
                    for(int k=j+1; k<strlen(str); k++) { //k is on str so its end on
                        length

                        if(sub[x]==str[k]) {
                            //get next match element so incremwnt sub
                            x++;

                            len++;//to check how much times condition
                            satisfied

                        } else {
                            break;
                        }
                    }
                    }//end of k for loop

                    if(strlen(sub)==len) {
                        return &str[j];

```

```

        }
    } //end of j for loop

} //end of i for loop

}

return NULL;
} else {
    return NULL;
}
}

```

```

//mystrrev
#include<stdio.h>
#include<string.h>
char* mystrrev(char*);
void main(){
    char str[5]="abcde";
    char* rev=mystrrev(str);
    printf("mystrrev :%s",rev);
}

```

```

char* mystrrev(char* str){
    int temp;
    int j=strlen(str)-1;
    for(int i=0;i<(strlen(str)/2);i++){

        temp=str[i];
        str[i]=str[j];

```

```

        str[j]=temp;

        j--;

    }

    return str;
}

//strchr (last occurrence)
#include<stdio.h>
#include<string.h>
char* mystrrchar(char*,char);
void main(){
    char str[]="ababncdn";

    char ch;

    printf("enter the char to(last occ) search:");
    scanf("%c",&ch);//b

    char* last_occ= mystrrchar(str,ch);
    printf("\nmystrrchr:%s",last_occ);
}

char* mystrrchar(char*str, char ch){
    //count
    int count=0;
    for(int i=0;str[i]!='\0';i++){
        if(str[i]==ch)
            count++;
    }

```

```

int num=count;

int k;

for(k=0;num!=0;k++){
    if(str[k]==ch){
        num--;

        //printf("ch :%c\n",ch);
        //printf("num:%d\n",num);
        printf("K:%d\n",k);
    }
}

// k++ (last iteration) k=8 then check num!='\0'
printf("K:%d\n",k-1);

//printf("str[k]:%c",str[k-1]);
return &str[k-1];
}

//mystrncmp
#include<stdio.h>
#include<string.h>
int mystrncmp(char*,char*,int);
void main(){
    char str1[5]="abcde";
    char str2[5]="absdc";

    int n ;

    printf("Enter the number of character wants to compare:");
    scanf("%d",&n);

    int res=mystrncmp(str1,str2,n);

```

```

        if(res==0)
            printf("same");
        else
            printf("not same");
    }

```

```

int mystrncmp(char* str1,char* str2,int n){

```

```

    for(int i=0;i<n;i++){
        if(str1[i]!=str2[i])
        {
            return 1;
        }
    }

```

```

    return 0;

```

```

}

```

```

//mystrncat

```

```

#include<stdio.h>

```

```

#include<stdlib.h>

```

```

#include<string.h>

```

```

char * mystrncat(char*,char*,int);

```

```

void main(){

```

```

    char str1[20]="prachiti";

```

```

    char str2[]="thakurFbs";

```

```

    int n ;

```

```

    printf("Enter the number of character wants to concat:");

```

```

    scanf("%d",&n);

```

```

        char* n_cat=mystrncat(str1,str2,n);
        printf("mystrcat:%s",n_cat);
    }

```

```

char * mystrncat(char* str1,char*str2,int n){

```

```

    if(strlen(str2)>=n){
        int len=strlen(str1);

        for(int i=0;i<n;i++){
            str1[len]=str2[i];
            len++;
        }

```

```

        return str1;
    }else{
        return NULL;
    }

```

```

}

```

```

//mystrncasecmp

```

```

#include<stdio.h>

```

```

#include<string.h>

```

```

int mystrncasecamp(char*,char*,int);

```

```

void main(){

```

```

    char str1[5]="abcde";

```

```

    char str2[5]="ABCfE";

```

```

    int n ;

```



```
printf("Enter the number of character wants to compare:");
```

```
scanf("%d",&n);
```

```
int res=mystrncasecamp(str1,str2,n);
```

```
if(res==0){
```

```
    printf("same");
```

```
}
```

```
else{
```

```
    printf("not same");
```

```
}
```

```
}
```

```
int mystrncasecamp(char* str1,char*str2,int n){
```

```
    int count=0;
```

```
    for(int i=0;i<n;i++){
```

```
        if(str1[i]==str2[i]){
```

```
            count++;
```

```
        }else{
```

```
            if((str1[i]==str2[i]-32) || str1[i]==str2[i]+32){
```

```
                count++;
```

```
            }
```

```
        }else{
```

```
            return 1;
```

```
        }
```

```
    }
```

```
}
```

```
    if(count==n){
```

```
        return 0;
```

```
    }
```

```
}
```

```
//strlwr
```

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

```
char* mystrlwr(char*);
```

```
void main(){
```

```
    char str[]="PRACHITI";
```

```
    char* lower=mystrlwr(str);
```

```
    printf("mystrlwr:%s",lower);
```

```
}
```

```
char* mystrlwr(char* s){
```

```
    for(int i=0;s[i]!='\0';i++){
```

```
        if(s[i]>='A'&& s[i]<='Z')
```

```
            s[i]=s[i]+32;
```

```
    }
```

```
    return s;
```

```
}
```

```
//mystrrchr
```

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

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

```
char* mystrrchr(char*,char);
```

```
void main(){
```

```
    char str[]="ababncdn";
```

```
    char ch;
```

```
    printf("enter the char to(last occ) search:");
```

```
    scanf("%c",&ch);//b
```

```

        char* last_occ= mystrrchar(str,ch);
        printf("\nmystrchr:%s",last_occ);
    }

```

```

char* mystrrchar(char* str,char ch){

    for(int i=(strlen(str)-1);i>=0;i--){
        if(str[i]==ch)
            return &str[i];
    }

    return NULL;
}

```

```

//strcpy

```

```

#include<stdio.h>

```

```

#include<string.h>

```

```

char* mystrcpy(char*,char*);

```

```

void main(){
    char str1[10]="prachiti";
    char str2[15];

    char* cpy=mystrcpy(str2,str1);
    printf("strcpy:%s",cpy);

}

```

```

char* mystrcpy(char* str2,char*str1){

    for(int i=0;str1[i]!='\0';i++){
        str2[i]=str1[i];
    }
}

```

```
}
```

```
str2[i]='\0';
```

```
return str2;
```

```
}
```

```
//strchr
```

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

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

```
char* mystrchr(char*,char);
```

```
void main(){
```

```
    //char array
```

```
    char str[10]="abdec";
```

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

```
    char ch;
```

```
    printf("enter the char to search:");
```

```
    scanf("%c",&ch);
```

```
    char* chr=mystrchr(str,ch);
```

```
    printf("strchr:%s",chr);
```

```
}
```

```
//function defination
```

```
char* mystrchr(char* str,char ch){
```

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

```
        if(str[i]==ch)
```

```
            return &str[i];
```

```

    }
}

//mystrcasecmp
#include<stdio.h>
#include<string.h>
int mystrcasecmp(char*,char*);
void main(){
    char str1[]="abcd";
    char str2[]="ABCe";

    int res=mystrcasecmp(str1,str2);
    if(res==0)
        printf("same");
    else
        printf("not same");

}

```

```

int mystrcasecmp(char* str1,char* str2){

    if(strlen(str1)==strlen(str2)){

        int count=0;
        for(int i=0;i<strlen(str1);i++){
            if(str1[i]==str2[i]){
                count++;
            }else{
                if(str1[i]==str2[i]-32){
                    count++;
                }
            }
        }
    }
}

```

```

        }else{
            if(str1[i]==str2[i]+32){
                count++;
            }
            else{
                return 1;//not match that
            }
        }
    }

    //after comparing whole string
    if(count==strlen(str1)){
        return 0;
    }

}

else{
    return NULL;
}

}

//mystrupr
#include<stdio.h>
char* mystrupr(char*);

void main(){
    char str[]="prachiti";
    char* s=mystrupr(str);

```

```

        printf("mystrupr: %s",s);
    }

char* mystrupr(char* str){
    for(int i=0;str[i]!='\0';i++){
        if(str[i]>='a' && str[i]<='z')
            str[i]=str[i]-32;
    }

    return str;
}

```

```

//mystrncpy
#include<stdio.h>

#include<string.h>

```

```

char* mystrncpy(char*,char*,int);

void main(){
    char str1[10]="Prachiti";
    char str2[10];
    int n;
    printf("Enter n number:");
    scanf("%d",&n);
    char* cpy=mystrncpy(str2,str1,n);
    printf("strncpy:%s",cpy);

}

```

```
char* mystrncpy(char* str2,char* str1,int n){
```

//here we cant check the len of str2 bcz str2 size is 10 length ==count of charc but now is nothing in str2

```
    if(strlen(str1)>=n){
```

```
        int i;
```

```
        for(i=0;i<n;i++){
```

```
            str2[i]=str1[i];
```

```
            //printf("i :%d",i);
```

```
        }
```

```
        str2[i]='\0'; //if we cant do this . then str2 will char array without null
```

```
        //so %s will not get null after end ->in main while printing
```

```
        return str2;
```

```
    }
```

```
    else{
```

```
        return NULL;
```

```
    }
```

```
}
```