

M Waldiodio D Ndiaye

Exercice 0:

```
#include <stdio.h>
#include <stdlib.h>
#define TAILLE_MAX 100

int main()
{
    char chaine[TAILLE_MAX];
    int i=0;
    printf("Entrez une chaine de caractere\n");
    gets(chaine);
    while(chaine[i]!='\0'){i++;}
    printf("%s contient %d caracteres ", chaine,i);
    return 0;
}
```

Exercice 1:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define TAILLE_MAX 100

int main()
{
    char chaine[TAILLE_MAX];
    int i;
    printf("Entrer une chaine de caractere\n");
    gets(chaine);
    for(i=0;chaine[i]!='\0';i++)
    {
        if((chaine[i]>='a')&&(chaine[i]<='z'))
        {
            chaine[i]=chaine[i]-32;
        }
        else
        {
            if((chaine[i]>='A')&&(chaine[i]<='Z'))
            {
                chaine[i]=chaine[i]+32;
            }
        }
    }
    printf("La conversion donne %s\n", chaine);
    return 0;
}
```

Exercice 2:Solution optimale

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define TAILLE_MAX 100
```

```

int main()
{
    char chaine[TAILLE_MAX];
    int i,tmp,a=0;
    printf("Entrer une chaine de caractere\n");
    gets(chaine);
    tmp=strlen(chaine)-1;
    for(i=0;i<=tmp/2;i++)
    {
        if(chaine[i]!=(chaine[tmp-i]))
            {a=1;break;}
    }
    if(a==0)
        printf("%s est un palindrome\n",chaine);
    else
        printf("%s n est pas un palindrome\n",chaine);
    return 0;
}

```

Exercice 2:

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define TAILLE_MAX 100

```

```

int main()
{
    char chaine1[TAILLE_MAX], chaine2[TAILLE_MAX];
    int i,tmp;
    printf("Enter une chaine de caractere\n");
    gets(chaine1);
    tmp=strlen(chaine1);
    i=0;
    while(i<=tmp-1)
    {
        chaine2[i]=chaine1[tmp-1-i];
        i++;
    }
    chaine2[i]='\0';
    tmp=strcmp(chaine1,chaine2);
    if(tmp==0)
        printf("%s est un Palindrome.\n", chaine1);
    else
        printf("%s n est pas un Palindrome.\n", chaine1);
    return 0;
}

```

Exercice 3:

```

#include <stdio.h>
#include <stdlib.h>
#define TAILLE_MAX 100

```

```

int main()
{

```

```

char x;
char chaine[TAILLE_MAX];
int i=0,cpt=0;
printf("Entrer une chaine de caractere\n");
gets(chaine);
printf("Entrer le caractere a chercher dans la chaine\n");
scanf("%c",&x);

while(chaine[i]!='\0')
{
    if(chaine[i]==x)
    {
        cpt++;
    }
    i++;
}
printf("la chaine %s contient %d fois le caractere %c",chaine,cpt,x);
return 0;
}

```

Exercice 3:

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define TAILLE_MAX 100

int main()
{
    char chaine1[TAILLE_MAX],chaine2[TAILLE_MAX];
    int i=0,a=0,tmp;

    printf("Entrer deux chaines de caracteres\n");
    gets(chaine1);
    gets(chaine2);

    if (strlen(chaine1)!= strlen(chaine2))
    { printf("%s et %s ne sont pas des miroirs\n",chaine1,chaine2); a=1;}
    else
    {
        tmp=strlen(chaine1);
        while(i<=tmp-1)
        {if(chaine1[i]!=chaine2[tmp-1-i])
            {a=1;break;}
            i++;
        }
    }

    if(a==0)
    {printf("%s et %s sont des miroirs\n",chaine1,chaine2);}
    else
    {printf("%s et %s ne sont pas des miroirs\n",chaine1,chaine2);}
    return 0;
}

```

Exercice 4:

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define TAILLE_MAX 100

int main()
{
    char chaine[TAILLE_MAX];
    int i=0,cpt=0,tmp;

    printf("Entrer une chaine de caracteres\n");
    gets(chaine);

    tmp=strlen(chaine);
    while(i<=tmp-1)
    {

if(chaine[i]=='a' || chaine[i]=='e' || chaine[i]=='o' || chaine[i]=='u' || chaine[i]=='i'
' || chaine[i]=='y' ||

chaine[i]=='A' || chaine[i]=='E' || chaine[i]=='O' || chaine[i]=='U' || chaine[i]=='I' ||
chaine[i]=='Y')
        {cpt++;}

        i++;
    }
    printf("%s contient %d voyelles\n",chaine,cpt);
    return 0;
}

```

Exercice 5:

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define TAILLE_MAX 100

int main()
{
    char chaine[TAILLE_MAX];
    int i,j,a,len,cpt,num=1,den=1;

    printf("Entrez une chaine de caractere\n");
    gets(chaine);
    //Il d'abord ramener toute la chaine soit au minuscule ou au majuscule
    //Pour que les caracteres repétés aient meme code ASCII sinon le M et m
    //De Matam sera vu comme deux lettres differentes.
    //commande pour mettre la chaine en minuscule
    strlwr(chaine);

    len=strlen(chaine);
    for(i=0;i<=len-1;i++)
    {
        num=num*(i+1);
        cpt=0;a=0;
        for(j=0;j<=len-1;j++)

```

```

        {
            if(chaine[i]==chaine[j]){cpt++;}
            if((chaine[i]==chaine[j])&&(j<i)){a=1;}
        }
    if((a==0)&&(cpt>=2))
    {
        for(j=1;j<=cpt;j++)
            {den=den*j;}
    }
}
printf("Le nombre d anagramme de %s est %d\n",chaine,num/den);
return 0;
}

```