

(18-19)

Algorithmique et structures de données1

E.M.D (durée 1h30)

Ex1 : Ecrire un programme C intelligent qui réalise le produit de 2 entiers positifs (l'opérateur de multiplication n'existe pas). (5 pts)

Ex2 : En supposant que la bibliothèque string.h n'existe pas, écrire un programme C qui lit une chaîne de caractères et affiche sa longueur. (5 pts)

Ex3 : Soient X et n des nombres entiers. Ecrire un programme C qui calcule X^n (X à la puissance n). (5 pts)

Ex4 : Soit une séquence de nombres entiers qui se termine par 999. (En aucun cas cette séquence dépasse 100 nombres). Ecrire un programme C qui calcule la moyenne arithmétique de cette séquence et le nombre de valeurs qui sont supérieures à cette moyenne. (5 pts)

(18-19)

Algorithmique et structures de données 1

Corrigé de l'E.M.D

Ex1 : #include <stdio.h>

```
main()
{int a, b, i, p=0;
printf("Donner 2 entiers positives : ");
scanf("%d %d",&a,&b);
if (a<b)
    for (i=1;i<=a;i++)
        p=p+b;
else
    for (i=1;i<=b;i++)
        p=p+a;
printf("le produit de %d par %d = %d ",a,b,p) ;
}
```

5

Ex2 : #include <stdio.h>

```
main()
{char mot[50], ch ; int L=0 ;
printf("Donner un mot : ") ; scanf("%s", mot) ;
do
{ch=mot[L] ;
L++ ;}
while (ch !='\0');
L-- ;
printf("La longueur du mot %s = %d",mot,L);
}
```

5

Ex3 : #include <stdio.h>

```
main()
{float p=1 ; int x,n,q,i ;
 printf("Donner la valeur de x :"); scanf("%d",&x) ;
 printf("Donner la valeur de n :"); scanf("%d",&n) ;
 if (n<0)
   q=-n ;
 else
   q=n ;
 for(i=1;i<=q;i++)
   p=p*x;
 if (n<0)
   p=1/p;
 printf(" %d à la puissance %d = %f",x,n,p) ;
 }
```

5

Ex4 : #include <stdio.h>

```
main()
{int T[100], i=0, nb, moy, val, supmoy=0, s=0 ;
 scanf("%d",&val);
 while (val != 999)
 {T[i]=val;
  i++;
  scanf("%d",&val);
 }
 nb=i;
 for (i=0;i<nb;i++)
   s=s+T[i];
 moy=s/nb;
 for (i=0;i<nb;i++)
   if (T[i]>moy)
     supmoy++;
 printf("le nombre de valeurs supérieures à la moyenne = %d",supmoy) ;
 }
```

5