Université Libanaise
Faculté des Sciences III
Tripoli – Liban

Cours: **I1101**
Examen : **Partiel**

Date : **2 /4/ 2019**
Durée : **1h**

Exercice I : (30 points)

Soit le code en C du programme suivant :

```
#include<stdio.h>
void main ()
   { float  d, i, s=0 ;
     int j =1;
    i=1;
    do{   d =i / j;
          s =s + d;
          j = j* 2;
          i = i +2;}
    while  ( i<= 7);
    printf (" la valeur est :  %f \n", s ) ;
   }
```

Faire (en détail) l'exécution de ce programme et donner son résultat .

Exercice II : (40 points)

Ecrire un programme qui lit un entier n tel que $4 < n \leq 10$ et qui dessine un schéma comme suit

```
* * * * * * * * * * * * * *
* * * * _ * * * * * * * *
* * * _ _ _ * * * * * *
* * _ _ _ _ _ * * *
* _ _ _ _ _ _ _ *
```

L'exemple est fait pour n = 5.

Exercice III : (30 points)

Ecrire un programme qui lit un entier $n > 5$ et qui calcule et affiche la somme suivante :

$$S_n = \sum_{k=1}^{n} (-1)^{K+1} K! = 1 - 2! + 3! - 4! + 5! - \ldots\ldots (-1)^{n+1} \times n!$$

Bon travail

Correction du Partiel :

Ex 1 :

$s = 0$; $j = 1$; $i = 1$ ⑥

1er $d = i/j = 1$; $s = 1$; $j = 2$; $i = 3$; $i <= 7$ ⑥

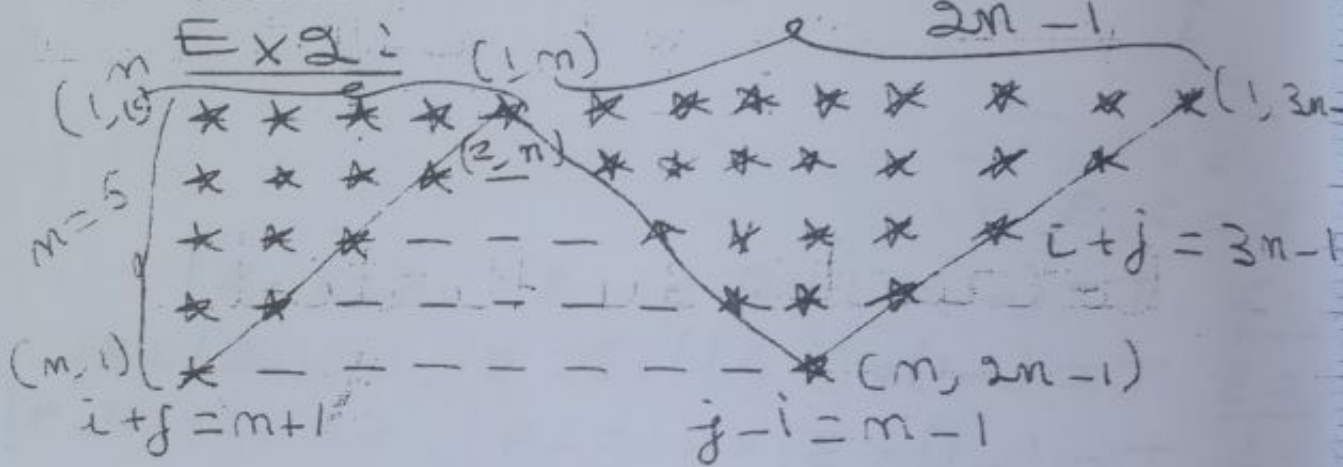
2ème $d = 3/2$, $s = 1 + \frac{3}{2}$, $j = 4$, $i = 5$, $i <= 7$ ⑥

3ème $d = 5/4$, $s = 1 + \frac{3}{2} + \frac{5}{4}$, $j = 8$, $i = 7$, $i <= 7$ ⑥

4ème $d = 7/8$, $s = 1 + \frac{3}{2} + \frac{5}{4} + \frac{7}{8}$, $j = 16$, $i = 9$, $i <= 7$ ⑥

la valeur est : ---- ⑥ X

Ex 2 :




```
#include <stdio.h>
void main()
{int i, j, n;                                   ②
 do{ printf ("donner n :");                     ④
     scanf ("%d", &n);}
 while(n<=4 || n>=10);                          ⑥
 for (i=1; i<=n; i++ )                          ⑤
  { for(j=1; j<=3n-1-i; j++)                    ⑤
    if(i+j > n+1 && j-i<n-1)                    ⑥
      printf ("_");                             ④
    else printf ("*");                          ④
  printf ("\n"); }                              ④
```

Ex 3 :

```
#include <stdio.h>
void main()
{int K, m; S=1; T=1;
do { printf("donner n>5:");
     scanf("%d", &m); }
while (m<=5);
for (K=2; K<=m; K++)
  { T=(-1)*T*K;
    S=S+T; }
  printf("la somme est: %d", S);
}
```

m=4

S=1, T=1.

K=2, 3, 4.

K=2 T=(-1)*1*2 = -2!, S=1-2!

K=3 T=(-1)*(-2!)x3 = 3!, S=1-2!+3!

K=4 T=(-1)x3!x4 = -4!, S=1-2!+3!-4!

Université Libanaise
Faculté des Sciences III
Tripoli – Liban

Cours: **I1101**
Examen : **Final +Partiel**

Date : **30 Juin 2018**
Durée : **2h et 30m**

Partie P (partiel)

Exercice I : (50 points)

Soit le code en C de la fonction suivante :

```
int  travail  (int  n)
    {  int  s, i=0 ;
       s=0 ;
       While  (n != 0)
         {    s = s +n % 10 ;
              n = n / 10 ;
              i ++ ;
         }
       printf (" nb = %d \n", i ) ;
       return  s ;
    }
```

- Exécuter la fonction travail (785402),
- Que fait cette fonction en général ?

Exercice II : (50 points)

Ecrire un programme qui lit un entier n tel que $4 < n < 15$ et qui dessine un trapèze comme suit :

```
* * * * * *
* * * * * * *
* * * * * * * *
* * * * * * * * *
* * * * * * * * * *
* * * * * * * * * * *
```

L'exemple est fait pour $n = 6$.

Partie F (final)

Exercice III : (20 points)

Ecrire un programme qui lit un entier positif n et qui calcule et affiche la somme suivante :

$$S_n = \sum_{k=0}^{n} \left(\frac{5k+1}{2(k*k)+3} \right)$$

Exercice IV : (20 points)

Quel résultat affiche ce programme

```
void main ()
  { int sum =0 ;
    int x =1 ;
    int i ;
    for (i=0; i<=4; i++)
    {    sum = sum +x;
         x = 2*x;
    }
   printf (" %d \n", sum );
}
```

Exercice V : (60 points)

On considère un tableau A [ ] qui contient 10 entiers. Ecrire les fonctions suivantes :

1. *lire_Tab(..:)* qui permet de saisir le tableau A [ ].
2. *Moyenne_Tab(...)* qui prend le tableau A [ ] comme paramètre et renvoie la moyenne des éléments de A [ ].
3. *teste_croissant(...)* qui prend le tableau A [ ] comme paramètre et retourne 1 si les éléments de A[] sont en ordre croissant et retourne 0 sinon ;
4. *renverse_Tab (..)* qui prend comme le tableau A[] comme paramètre et renverse ses éléments c.-à-d.le premier élément devient le dernier et le dernier devient le premier et ainsi de suite (sans utiliser un deuxième tableau); *Ex : si A = {0,1.2,3,4,5}* le tableau demande devient *A ={5,4,3,2,1,0} ;*
5. *remplace(..:)* qui prend comme le tableau A[] comme paramètre et remplace chaque élément négatif par le nombre des éléments qui le précèdent ;
   Ex : si A= {3, 1, -4, 2, -8} le tableau devient A= {3, 1, 2, 2, 4} ;
6. *repetition_max(...)* qui prend le tableau A comme paramètre et retourne le nombre de répétition de l'élément maximum de A ;

Ecrire un programme en C qui fait appel aux fonctions précédentes et affiche leurs résultats.

2017 - 2018 1ère session I1101

Partie Final :

Ex III :

```
#include <stdio.h>
void main(){
int i ;
float s=0;
do { printf("donner n+");
     scanf("%d",n);
   }
while (n<=0)
for(i=0, i<n, i++)
s = s+ ((5*i)+1) *1.0)/((2*i*i)+3)
printf("le somme est %f", s);
}
```

Ex V :

(1)*
```
void lire-tab (int V[])
{int i;
 for(i=0; i<10; i++)
{printf("donner V[%d]", i);
 scanf("%d", &V[i]); }
}
```

(2) *
```
float moyenne (int t[])
{int i, s=0;
 for(i=0; i<10; i++)
  s=s+t[i];
return (float)s/10;}
```

(3)* int teste-croissant(int t[])
{int i, ok=1;
for(i=0; i<9 ; i++)
if(t[i]>t[i+1]) ok=0;
return ok;}

2ème Méthode

{int i = 0;
while(t[i]<=t[i+1] && i<9)
i++;
if(t[i]>t[i+1]) return 0;
else return 1;
}

(4)* void reverse(int A[])
{int i, M;
for(i=0; i<5; i++)
{M = A[i];
A[i]= A[9-i];
A[9-i]= M;}
}

(5)* void remplace(int T[]){
int i;
For(i=0; i<10 ; i++)
if(T[i]<0) T[i]= i;
}

(6)* int répétition(int T[])
{int i, M=t[0], K=1;
For(i=0, i<10, i++)
if(T[i]>M){M= t[i]; K=1;}
else if(t[i]==M) K++;
return K;}

progra e principal

```
#include <stdio.h>
  void main()
{ int T[10];
 lire_tab(t);
 if (Teste_croissant(t) == 1)
  printf("les élts sont en ordre
          croissant \n" );
  else
    printf("les élts ne sont pas en ordre
          croissant \n"
 renverse(t)
 printf("la moyenne est: %f \n", );
          moyenne(t)
 remplace(t);
  printf("répétition de max est: %d\n",)
        répétition_max(t)
 }
```

niversité Libanaise - Faculté des Sciences III
épartement Mathématiques-Informatique
OURS : I1101
ession : partiel

Date : 12 / 04 / 2017
Durée : 1h

xercice I (30 points)

Ecrire un algorithme (ou un programme en C) qui demande à l'utilisateur de saisir un entier n. Cet algorithme doit calculer et afficher le tableau suivant:

1	2	3	....	n
F(1)	F(2)	F(3)	.....	F(n)

Où $F(i) = \sum_{k=1}^{k=i} k$

Exercice II : (30 points)

Ecrire un programme qui lit un entier n et qui dessine un triangle (de n lignes) comme suit:

pour n = 4

```
*******
 *****
  ***
   *
```

pour n = 6

```
***********
 *********
  *******
   *****
    ***
     *
```

Exercice III : (40 points)

Ecrire un programme en C qui lit un entier n. Ce programme doit lire une série de n entiers et calculer et afficher :

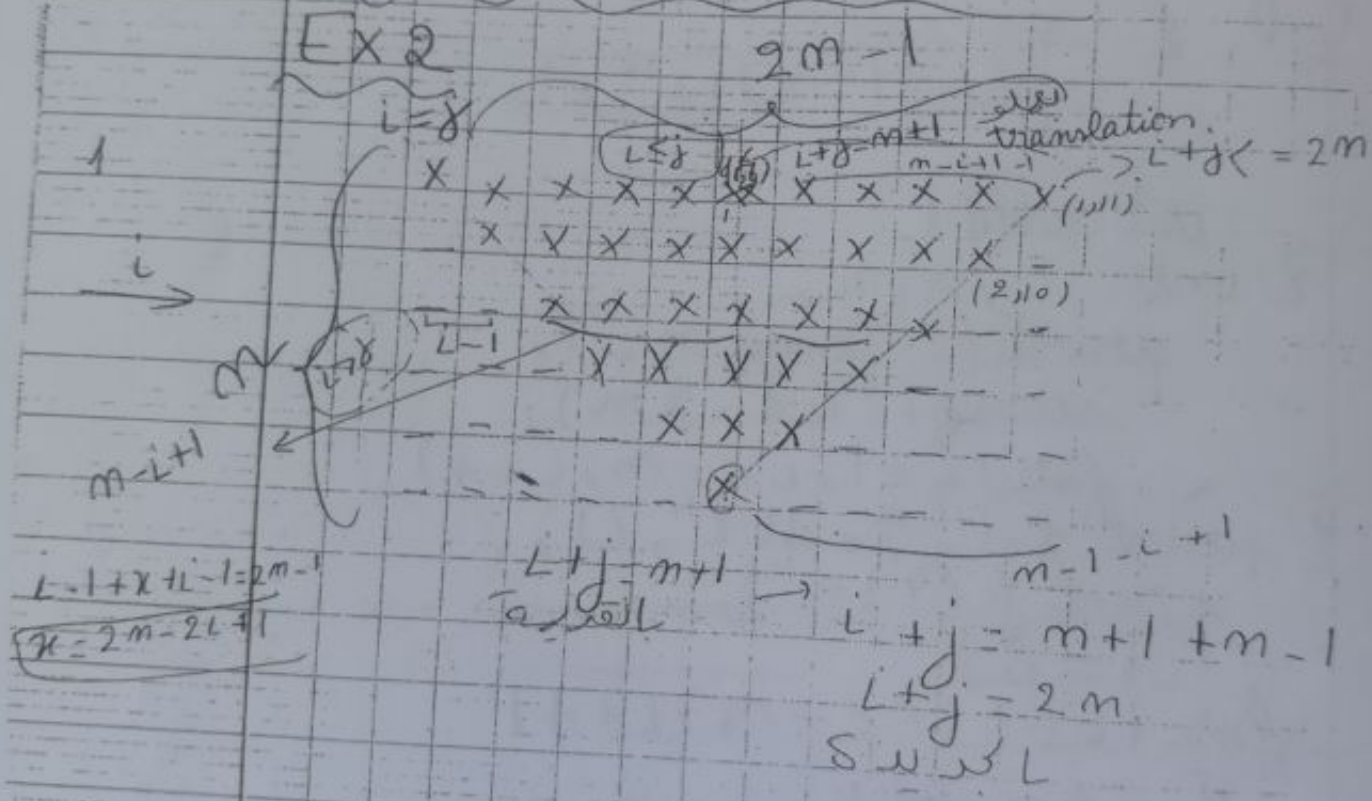
- la somme des entiers positifs ainsi leur moyenne,
- la somme des entiers négatifs ainsi leur moyenne.

Bon travail

autre méthode

```
for (i=1; i<=n; i++)
{ s=0;
  for (j=1; j<=i; j++)
    s = s+j ;
  printf("%d\t", s);
}
```

EX 2



$i-1+x+i-1=2m-1$

$x=2m-2i+1$

$i+j=m+1+m-1$

$i+j=2m$

```
int void()
{ int i, j, n;
  printf("donner n:");
  scanf("%d", &n);
  for(i=1; i<=n; i++)
  { for(j=1; j<=2*n-1; j++)
```

```
if (i <= j && i + j <= 2 * n)
    printf("*");
  else printf("_");
 printf("\n");
 }
 return 0
}
```

autre mét

```
for (i=1; i<=n; i++)
{ for (j=1; j<=i-1; j++)
    printf("_");
  for (j=1; j<=2*n-2*i+1; j++)
    printf("*");
```

autre mét

n - i
n - i + 1
2n - 2i + 1

```
  for (j=1; j<=n-i+1; j++)
    Printf("*");
  for (j=1; j<=n-i; j++)
    Pritf("*");
```

Ex III

```
int main ()
{ int n, i, sp, sn, cp, cn, e;
  float m1, m2;
  printf ("Donner n : ");
  scanf ("%d", &n);
  if (n <= 0) printf ("Erreur");
  else {
      sp = 0;
      sn = 0;
      cp = 0;
      cn = 0;
  for (i = 1; i <= n; i++)
      printf ("Donner e : ");
      scanf ("%d", &e);
      Pair ou impair (e%2 == 0)
  if (e >= 0) { sp = sp + e; cp++; }
  else { sn = sn + e; cn++; }
  m1 = ((float) sp) / cp;
  m2 = ((float) sn) / cn;
  printf ("la somme est : sp = %d", sp);
  printf ("sn = %d", sn);
  printf ("moy >= 0 : %f", m1);
  printf ("moy < 0 : %f", m2);
  return 0
}
```

```
if (i <= j && i + j <= 2 * n)
    printf("*");
  else printf("_ ");
 printf("\n");
 }
 return 0
}
```

autre mét

```
for (i=1; i <= n; i++)
{ for (j=1; j <= i-1; j++)
    printf("_ ");
  for (j=1; j <= 2*n - 2*i + 1; j++)
    printf("*");
```

autre mét

n - i
n - i + 1
2n - 2i + 1

```
for (j=1; j <= n-i+1; j++)
    Printf("*");
  for (j=1; j <= n-i; j++)
    Pritf("*");
```

Ex.1

1	2	3	4	...	n
F(1)	F(2)	F(3)	F(4)	...	F(n)

$$F_n = \sum_{1}^{n} i = \sum_{1}^{n-1} i + n = F(n-1) + n$$

```
for (i=1;i<=n;i++)
   printf(" *\t"); {printf("%d\t",i);}
                                     valeur
            (espace, itération)
                    printf("\n");
s = 0;
for (i=1;i<=n;i++)
  { s = s+i;
    printf(" %d\t",s);

int void ( )
{ int i, s, n;
   printf("Donner n : ");
   scanf("%d", &n);
   for (i=1;i<=n;i++)
{ printf (" %d\t", i);
  printf ("\n");
   s = 0;
for (i=1;i<=n;i++)
 {  s = s+i
   printf ("%d\t",s)
}
```

autre méthode

suite ari $S = \frac{n(n+1)}{2}$

```
~~s=0;~~
for (i=1;i<=n;i++)
{  ~~s = s+i;~~
   printf("%d\t", i*(i+1)/2);
```