Test1 ASDD 2022-2023 1h plus corrigé

Exercice 1 :

1) Soit la fonction mathématique de collatz définie sur les nombres entiers naturels comme suit :

$$f(n)=\begin{cases} n/2 \text{ si n est pair} \\ 3*n+1 \text{ sinon} \end{cases}$$

Ecrire une fonction qui permet de calculer f(n).(0.5pt)

2) soit la suite de Syracuse définie comme suit :

$$\begin{cases} U_0 = n \\ U_{n+1} = f(U_n). \end{cases}$$

Sachant que cette suite converge vers la valeur 1 quelle que soit la valeur de $n \in N^*$
Donner la valeur des différents termes de cette suite pour $U_0=3$ puis pour $U_0= 4$ puis pour $U_0=6$.(1.5pt)

3) Ecrire la fonction récursive Syracuse qui ne retourne pas de résultats et calculant les termes de cette fonction avec l'instruction qui permet d'afficher les valeurs de chaque terme pour une valeur de départ n. (1.5pt)

4) Modifier le code de Syracuse en une fonction récursive qui renvoie le nombre d'applications de la fonction de Collatz nécessaires pour atteindre 1. (1.5pt)

Exercice 2 :

1) Ecrire une fonction qui vérifie si une liste chainée d'entiers est ordonnée.(2pts)
2) Sachant que la liste est ordonnée écrire la fonction qui supprime les valeurs redondantes.(3pts)

Corrigé Test1 ASDD 2022-2023

Exercice 1:

1)(0.5pt)

```
int  f (int n)
{
if (n%2==0)  return n/2;
return 3*n+1;
}
```

2) a-(0.5pt)

U0=3
U1=f(U0)=f(3)=10
U2=f(U1)=f(10)=5
U3=f(U2)=f(5)=16
U4=f(U3)=f(16)=8
U5=f(U4)=f(8)=4
U6=f(U5)=f(4)=2
U7=f(U6)=f(2)=1  donc on le dernier terme à déterminer est U7
Puisque  Si on calcule U8=f(U7)=f(1)=4 on retrouve le cas la valeur de U5 donc une boucle infinie pour valeurs (4,2,1) donc on dit converge vers 1.

b-U0=4
U1=f(4)=2 puis U2=1 (0.5pt)

c- U0=6
U1=f(6)=3 donc avec U1 on retombe sur le cas  a-  (0.5pt)

3)  fonction qui affiche les termes de la fonction syracuse

```
void syracuse( int  n)
{
     if(n>1) { printf("%d\n", n ) ;     (0.5pt)
               syracuse(collatz(n)) ;   (0.5pt)
              }
     else  printf("%d\n",1);     (0.5pt)
}
```

4) fonction qui compte le nombre d'appels de la fonction collatz

```
int syracuse2( int  n)
{      if(n>1) { printf(" %d\n",n ) ;
                 return 1+ syracuse2(collatz(n)) ; (1pt)
               }
       printf(" %d\n",n );
       return 0 ;      (0.5pt)
}
```

**********

**Bonnus :** Ici une manière d'afficher les termes de Syracuse.

```
void syracuse( int i, int  n) // i numéro du terme dans main i=0 de U0
{     if(n>1) { printf("U%d = %d\n",i,n ) ;
                syracuse(i+1,collatz(n)) ;
               }
     else printf("U%d = %d\n",i,n );      //dernier terme quand n devient 1
}
main(){  syracuse( 0,15);}
```

**********

Exercice 2 :

1)

```
int verif_liste_croissante(liste*t)  //fonction qui vérifie l'ordre
croissant
{ if(t==NULL) return -1; (0.5pt)
    else {liste*p=t->suivant; (0.5pt)
          while(p!=NULL && t->val<=p->val) (0.5pt)
            {t=p; p=p->suivant;
            }
           if (p==NULL) return 1;  (0.5pt)
           else return 0;
          }
}
```

2)

```
void supredondance(liste*t)
{ liste *p,*k;
        p=t;
        if (p!=NULL)  (0.25pt)
          {liste *suiv= p->suivant;
            while (p!=NULL && suiv!=NULL) (0.5pt)
            {
                if (p->val == suiv->val)
                 {k=suiv;
                  p->suivant=suiv->suivant;   (1.25pt)
                  suiv=suiv->suivant;
                  free(k);
                 }
                 else { p=suiv;
                        suiv=suiv->suivant;   (1)
                      }
             }
            }
}
```