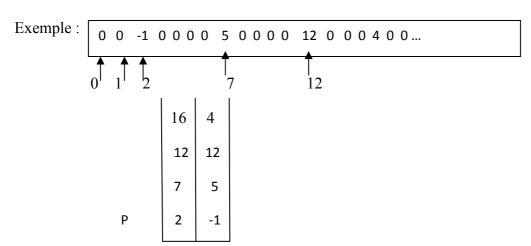
Test2 ASDD 2022-2023

1h

Exercice 1: (4pts)

Etant donné un tableau de données de grande taille N dont la plus part de ses valeurs sont égales à 0, on désire ne pas stocker ces 0 dans notre programme en définissant une pile P de structures qui représente la compression de ces données. Les éléments de P représentent l'indice et la valeur de chaque donnée différente de 0.



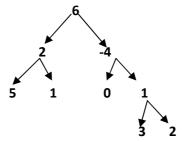
Ecrire les codes de programme C:

- 1) déclarer la structure statique de pile qui permet de représenter les données du tableau
- 2) une fonction qui permet de lire des données et de les représenter dans la pile.
- 3) une fonction qui permet de saisir un numéro d'indice d'une donnée et d'afficher sa valeur. (exemple : le numéro d'indice égale à 1 sa valeur est 0 et le numéro d'indice égale à 7 sa valeur est 5....).

Exercice 2 (6pts)

Soit à initialiser dans un programme C l'arbre binaire suivant :

1) donner la structure de données et le code des instructions qui permettent de représenter l'arbre suivant et les codes de:



- 2) la fonction récursive qui permet de calculer la somme des valeurs de cet arbre.
- 3) la fonction itérative équivalente à la fonction somme.
- 4) la fonction qui cherche une valeur val et retourne le niveau où elle se trouve.

Corrigé test 2 ASDD 2022-2023

ASDD

```
Exercice 1:4pts
1) 0.5pt
Structure de la valeur de la pile
#define maxtaille 100
typedef struct donnee
{ int val;
   int indice;
}donnee;
Typedef struct pile
{ donnee don[maxtaille];
   int ind;
}pile ;
2) 1.5pt
Void lecturetab(pile *p, int n)
{donnee x ;int i ;
  for(i=0 ;i<n;i++)</pre>
  {printf(" tab[%d]=",i); x.indice=i;
   scanf(("%d\n",&x.val);
   empiler(p,x);
  }
}
3) 2pts
Void affiche (pile p)
{pile pi ; initpile(&pi) ; donnee x ; int i ;
  printf((" l'indice ?");scanf("%d\n",&i);
 while(!pilevide(p))
     { depiler(&p,&x);empiler(&pi, x);
       if( x.indice ==i)
         {printf(" la valeur de tab[%d]=%d", i,x.val );break;}
      else if (x.indice <i) {printf(" la valeur de tab[%d]=0", i); break;}</pre>
if(pilevide(p)) printf(" la valeur de tab[%d]=0", i);
while(!pilevide(pi))
{depiler(&pi,&x);empiler(&p, x);}
}
```

ASDD

```
Exercice 2:6pts
1) 0.5
  typedef struct Noeud
         { int val;
            struct Noeud *FG, *FD;
         } Noeud;
 main()0.5
{ Noeud *racine=NULL;
     racine= cree_noeud (6);
     racine->FG = cree noeud (2);
     racine->FD = cree_noeud (-4);
     racine->FG->FG = cree_noeud (5);
     racine->FG->FD = cree noeud (1);
     racine->FD->FG = cree noeud (0);
     racine->FD->FD = cree noeud (1);
     racine->FD->FG = cree_noeud (3);
     racine->FD->FD->FD = cree_noeud (2);
 2) 1pt
int som(Noeud *a)
 {if (a == NULL) return 0;
  return a->val + som ( a->FG) + som ( a->FD);
 }
3)2pts
int som(Noeud *a)
 {pile P ; initpile(&p) ; som=0 ;
    empiler(&p, a);
       while(!pilevide(p))
          { depiler(&p,a);
              While(a!=NULL)
                  {som=som+ a-> val;
                   empiler(&p, a->FD);
                   a=a->FG;
                  }
          }
    return som;
 }
4) 2pts
Première solution
int rech niv ( Noeud *a,int v,int niveau)// dans main à l'appel niveau=0 ou 1
  { int x,y;
    if (a==NULL)return -1;
    if( a->val== v) return niveau;
    x=rech_niv ( a->FG, v, niveau+1);
    y=rech_niv ( a->FD, v, niveau+1);
    if(x>y && y!=-1)return y;
    if (x!=-1) return x;
    return y;
   }
4) 2pts
```

Deuxième solution avec le parcours par niveau vu en td algorithme itératif

```
int rech_niv ( Noeud *a,int x)
{if (a == NULL) return -1;
       else {FileNoeud fil1; Fileint fil2;
              enfilerN(&fil1,a) ;enfilerI(&fil2,0); int niveau ; // niveau 0 ou 1 ^{\circ}
              while( !filevideN(fil1) )
               {defilerN(&fil1,a); defilerI(&fil2,&niveau);
                if( a->val== x) return niveau;
                if( a->FG!=NULL) { enfilerN(&fil1,a->FG) ;
                                   enfilerI(&fil2,niveau+1);}
                if( a->FD!=NULL) { enfilerN(&fil1,a->FD) ;
                                   enfilerI(&fil2,niveau+1);}
               }
   return -1;
}
```

ASDD