

## Faire le point N°5 (benjamin Delpech)

--QUESTION 1-----

```
type t_cell;
type t_pt_Cell is access T_Cell;
type t_Cell is record
  Car1:Character;
  Car2:Character;
  Suiv:T_Pt_Cell;
end record;
```

-----  
--Question2-----

```
procedure Supprime_Non_Conforme(tete:in out t_pt_Cell)is
begin
  while tete.Suiv /= null loop
    case tete.Car1 is
      when 'A'|'C'|'T'|'G'|'a'|'c'|'t'|'g'=> null;
      when '_' => if tete.car2 = '_' then Supprimer(tete);end if;
      when others => Supprimer(tete);
    end case;
    case tete.Car2 is
      when 'A'|'C'|'T'|'G'|'a'|'c'|'t'|'g'=> null;
      when '_' => if tete.car1 = '_' then Supprimer(tete);end if;
      when others => Supprimer(tete);
    end case;

    tete:= tete.suiv;
  end loop;
  end Supprime_Non_Conforme;
```

-----  
--Question3-----

```
function Compte_Match(tete:t_pt_Cell)return Integer is
begin
  Tete.Car1:=To_Upper(Tete.Car1);
  Tete.Car2:=To_upper(tete.Car2);

  if Tete /= null then return 0;
  elsif L.Car1 =L.Car2 then
    return Compte_Match(tete.suiv);
  end if;
end Compte_Match;
```

-----  
--Question4-----

```
procedure Affiche_Seq_Ori(Tete:T_Pt_Cell)is
  tmp:T_Pt_Cell:=Tete;
begin
  if (Tmp /= null) and (Tmp.Car1/= '_') then
    Put(Tmp.Car1);

    if Tmp.suiv /= NULL Then
      Affihe_Seq_Ori(Tmp.suiv);
    end if;
  end if;
```

```

        end if;

    end Affiche_Seq_Ori;
    -----

---Les arbres-----

--a)---
type t_noeud;
type t_arbre is access T_noeud;
type t_noeud is record
    val:integer;
    fg,fd:T_arbre;
end record;
-----

-----

T_Arbre:=new T_Noeud' (1,Fg,Fd);
Fg:=new T_Noeud' (3,NULL,4);
Fd:=new T_Noeud' (5,null,null);

-----

-----

Parcours Préfixe:2,7,8,4,22,13,15,4,5,3,6,1,33,0,9,10,11,12,40,44,50
Symétrique:17,8,13,22,15,7,4,3,5,6,1,2,0,33,9,11,10,40,12,50,44
Suffixe:17,8,13,15,22,7,3,5,6,1,2,0,9,33,11,10,40,12,50,44

-----

-----

La Taille De L'Abre Est 21, Car On A 21Noeuds.
Sa Profondeur Est De 6.
La Hauteur Du Neoud De Clé 6 Est De 4.
Sa Longueur De Cheminement Est De 21, Car C'Est La Somme Des Noeuds.

-----

L'Arbre Est Vide-> A = null;
Arbre Ne Contient Qu'Une Racine -> A/= null and A.Suiv = null;
B Pointe Sur Un Noeud Interne Double -> B.Fg /= null and B.Fd /= null;
C Pointe Sur Un Fils De B -> C = B.Fg or C = B.Fd;

-----

-----

function Compte_Noeud_Term4(A:T_Arbre) return Integer is
    tmp:T_Arbre:=A;
begin
    if tmp = null then return 0;
    else
        if(tmp.Val mod 4 = 0) then return 1;end if;
        Compte_Noeud_Term4(Tmp.Fg);
        Compte_Noeud_Term4(Tmp.Fd);
    end if;
end if;

end Compte_Noeud_Term4

```