Les méthodes de tri

Sommaire

1) ***Tri à bulles***

Principe

- Comparer le 1er et le 2eme élément , s'ils ne sont pas dans le bon ordre alors on les permute
- Comparer le 2er et le 3eme élément , s'ils ne sont pas dans le bon ordre alors on les permute
- Répéter la comparaison ieme élément et le i+1eme élément, s'ils ne sont pas dans le bon ordre alors on les permute jusqu'a arriver a l'avant dernier élément
- Apres le 1er parcours total du tableau le plus petit(le plus grand)élément est a la 1ere position qui est sa position définitive
- Refaire le même traitement en s'arrêtant soit il n'y a plus de permutation durant le parcours (tableau trie) soit le traitement est au niveau de l'avant dernier et le dernier élément

Analyse du module de tri à bulles (Tri en ordre croissant)

```
DEF PROC triBulles(var T : Tab ; n : entier)
Resultat = T_trie
  T_trie= [ ] = Repeter
                        [b ← faux ] =
                        pour i de 1 a n-1 faire
                                si T[i]>T[i+1] alors
                                        Proc permut(T[i], T[i+1])
                                        b ← vrai
                                finSi
                        finPour
                        n ← n - 1
                Jusqu'a ( n = 1) ou (b = faux )
Fin triBulles
```

T.D.O.L

Objet	Nature/type
b	Var/booleen
i	Var/entier
permut	procédure

Algorithme ***de la*** de la procédure triBulles (***Tri en ordre croissant***)

```
0) DEF PROC triBulles(var T : Tab ; n : entier)
 1)  Repeter
                        [b ← faux ] =
                        pour i de 1 a n-1 faire
                                si T[i]>T[i+1] alors
                                        Proc permut(T[i], T[i+1])
                                        b ← vrai
                                finSi
                        finPour
                        n ← n - 1
                Jusqu'a ( n = 1) ou (b = faux )
2) Fin triBulles
```

Algorithme de la de la procédure permut (***Permutation de deux entiers***)

```
0) DEF PROC permut(var a,b : entier)
 1) c ← a
 2) a ← b
 3) b ← c
 2) Fin permut
```

Traduction pascal de la de la procédure triBulles (***Tri en ordre croissant***)

```
Procedure triBulles(var T : Tab ; n : integer) ;
 Var b: boolean ; i : integer ;
 Procedure permut (var a , b : integer);
  Var c : integer;
 begin
  c := a; a:=b ; b:= c ;
end;
begin
     repeat
          begin
                        b :=  false;
                        for i := 1 to  n-1 do
                                if T[i]>T[i+1] then
                                        permut(T[i], T[i+1]);
                                        b :=  true;
                     n := n - 1;
           end;
      until ( n = 1) or (b = false );
end;
```