

# Correction de devoir de contrôle n°1-2

## Analyse du programme principal

Résultat= T1, T2, T3

```
PROC affichage(T1, p)
PROC affichage(T2, q)
PROC affichage(T3, r)
PROC transfert (T, N, Cer)
PROC saisir_cercle(cer)
PROC saisir_tab (T,N)
```

## Analyse de la procédure saisir\_cercle

DEF PROC saisir\_cer( VAR cer :cercle)

Résultat= Cer

**Avec Cer faire**

```
Rayon=Donnée (“Taper le rayon :”)
Centre.x=donnée (“Taper l’abscisse du centre :”)
Centre.y=donnée (“Taper l’ordonnée du centre :”)
```

**Fin Avec**

## Analyse de la procédure saisir\_tab

DEF PROC saisir\_tab (VAR T :TAB ; VAR N :entier)

Résultat= T, N

T= [] **Pour i de ‘A’ à CHR(64+ N) Faire**

**Avec T[i] Faire**

```
X= donnée (“Taper l’abscisse :”)
Y= donnée (“Taper l’ordonnée :”)
```

**Fin Avec**

**Fin pour**

N=[] **Répéter**

N= donnée (“Taper le nombre de points :”)

**Jusqu’à N dans [3..26]**

## Analyse de la procédure transfert

DEF PROC transfert (VAR T1,T2,T3 :TAB2 ; VAR p,q,r :entier ; T :tab ; N :entier ; Cer :cercle)

Résultat= T1, T2, T3, p, q, r

[p←0 ; q←0 ; r←0]

**Pour i de ‘A’ à CHR(64+ N) faire**

**Si FN distance (t[i], cer.centre)=rayon Alors**

```
P←p+1
T1[p]←T[i]
```

**Sinon Si FN distance (T[i],cer.centre)< rayon Alors**

```
Q←q+1
T2[q]←T[i]
```

**Sinon**

```
R←r+1
T3[r]←T[i]
```

**Fin si**

**Fin pour**

## Analyse de la procédure affichage

DEF PROC affichage (T1:TAB2 ; p :entier )

Résultat= T1

T1=[] **Pour j de 1 à p faire**

Ecrire (T1[j].x, T1[j].y)

**Fin pour**

T.D.N.T

TYPE

Point= Enregistrement

X,y :réel

Fin point

Cercle= Enregistrement

Centre :point

Rayon :réel

Fin cercle

TAB= tableau de 26 point { indice caractères }

TAB2=tableau de 26 point {indice entier }



## Analyse de la fonction distance

DEF FN distance (a,b :point) :réel

Résultat= distance

Distance← Racine\_carré( carré(a.x-b.x) + carré (a.y-b.y) )