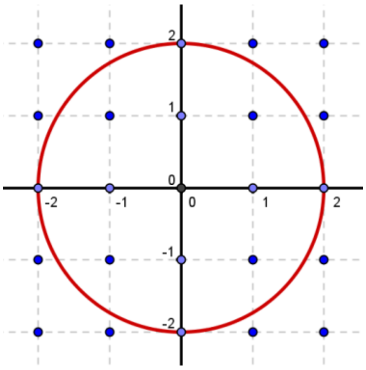
*Gérard CORDES – groupe TraAM Maths et TICE  de l'académie de Nantes – Mai 2012*

## Comptons les points (1S ou approfondissement en Seconde)




### *Compétences mathématiques :*

Il s'agit de la résolution d'un problème ouvert au cours de laquelle les élèves sont conduits à identifier le caractère inopérant de tout ce qu'ils connaissent sur les suites : le modèle mathématique n'est ni une suite géométrique, ni une suite arithmétique, ni une suite définie par une relation de récurrence.
▫ Le dénombrement demandé nécessite l'élaboration d'une démarche algorithmique.
▫ Mobilisation pertinente de l'outil algorithmique pour effectuer un comptage de points à coordonnées entières dans un disque.
▫ Suite définie géométriquement dont les termes peuvent être calculés par un procédé algorithmique.
▫ Problème de seuil pour une suite croissante.

### *Descriptif rapide*

A partir d'une situation concrète, l'élève découvre qu'un traitement algorithmique s'impose pour le comptage des points à l'intérieur d'un disque. Ce comptage, associé à l'utilisation d'un quadrillage, permet d'approcher l'aire d'un disque ou le nombre π.

### *Sommaire :*

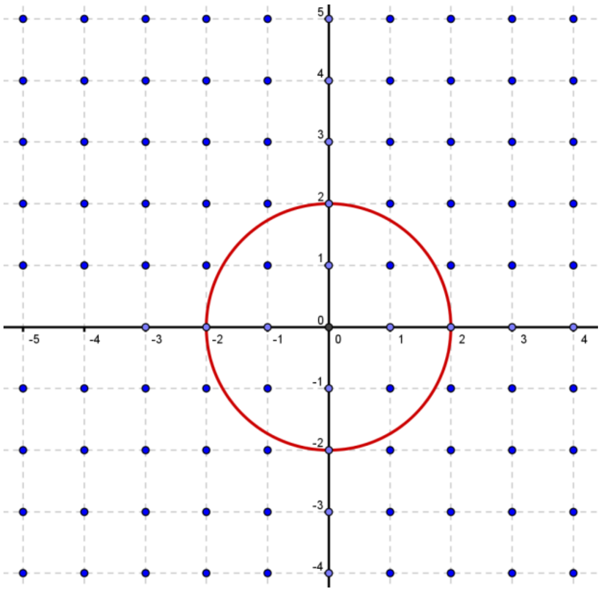
## *Objectifs du programme*

*Mettre en œuvre une recherche de façon autonome, mener des raisonnements, avoir une attitude critique vis-à-vis des résultats, utiliser l'algorithmique.*
*Cette activité sous forme ouverte vise prioritairement à renforcer la maitrise des compétences de résolution de problèmes. Elle permet l'utilisation incontournable d'une démarche algorithmique.*

## *Enoncé de l'exercice*

*Séance de travaux dirigés en seconde (pour les questions 1 et 2) ou première S.*

Le plan est muni d'un repère orthonormé (O ;I ;J) d'unité 1cm.
On se donne un entier naturel $n$ non nul.

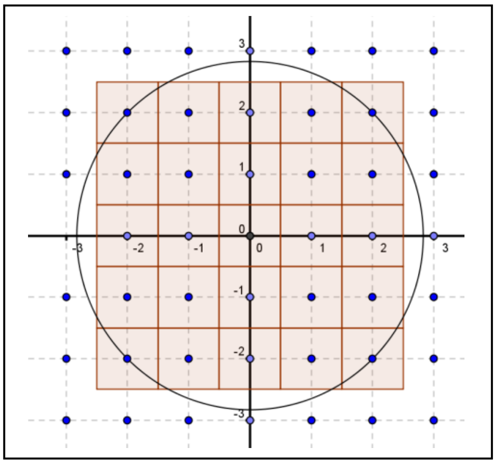


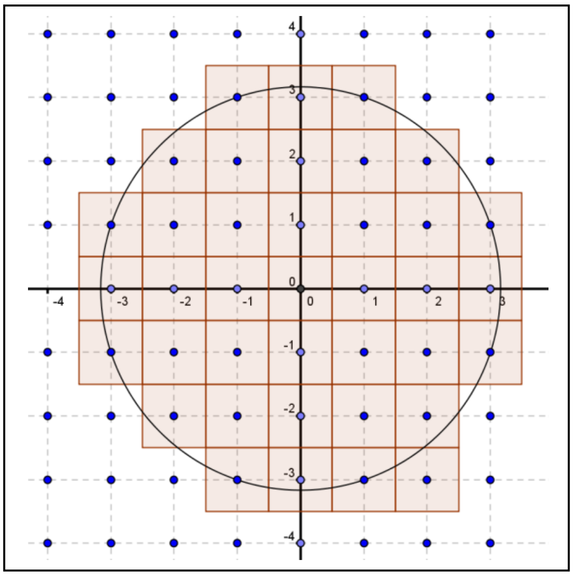
On se propose compter le nombre de points à coordonnées entières contenus dans le disque de centre O et de rayon $\sqrt{n}$ .

1. Imaginer une méthode de calcul qui permette de calculer le nombre de points à coordonnées entières contenus dans le disque de centre O et de rayon $\sqrt{n}$ .

2. Comment choisir l'entier $n$ pour que le disque de centre O et de rayon $\sqrt{n}$ contienne au moins 1000 points à coordonnées entières ?

3. Les figures suivantes obtenues pour $n$=8 et $n$=10 permettent de considérer que, pour les grandes valeurs de $n$, l'aire du disque exprimée en cm$^2$ est voisine du nombre de points à coordonnées entières contenus dans le disque de centre O et de rayon $\sqrt{n}$ .





Utiliser la méthode de comptage des points précédent pour donner une évaluation grossière du nombre π en choisissant *n* assez grand (par exemple *n*=1000).

## *Scénario :*

▪ Ce qui a été fait avant
Les élèves ont déjà élaboré un algorithme permettant de dire si un point est ou n'est pas dans une zone délimitée par deux courbes (activité présentée en annexe). Les élèves ont donc déjà rencontré une stratégie reposant sur deux tests successifs : l'un sur les abscisses et l'autre sur les ordonnées.

▪ ***Le vécu des élèves***

▫ La situation est vite comprise et le comptage à la main donne :

| | | | | | |
|---|---|---|---|---|---|
| *n* | 1 | 2 | 3 | 4 | 5 |
| *S* | 5 | 9 | 9 | 13 | 21 |

▫ Les élèves ont tout d'abord cherché un procédé logique pour compléter le tableau. Leur incapacité à en trouver les a momentanément bloqués. Ils découvrent qu'un traitement algorithmique s'impose pour le comptage des points à l'intérieur d'un disque. Au départ, ils ont eu du mal à concevoir l'algorithme mais l'idée de procéder par double balayage (d'abord suivant les valeurs de *x* puis suivant celles de *y* comme dans une activité déjà faite en classe voir en annexe) a été amenée par un groupe d'élèves.
L'idée s'est vite propagée. La confrontation de différentes stratégies conduit à deux idées fortes:
→ mettre l'origine à part pour éviter de compter ce point plusieurs fois.
→ procéder à un double balayage d'abord suivant les valeurs de *x* puis suivant celles de *y*. Le calcul a pu être optimisé en découpant le disque en quatre quarts de disque (dans chaque quart de disque l'un des bords droits est pris et l'autre non)

▫ L'utilisation du quadrillage permet d'approcher l'aire d'un disque ou le nombre π. Le professeur intervient alors pour montrer graphiquement comment un pavage du disque peut conduire à une évaluation grossière de l'aire de ce disque. On peut imaginer que cette idée peut aussi servir de situation déclenchante pour entrer dans la tâche.

▫ Le problème de seuil a été vite traité par un algorithme (on trouve $n \geq 320$).

▫ La recherche d'une approximation de π a suscité de l'intérêt et des recherches.

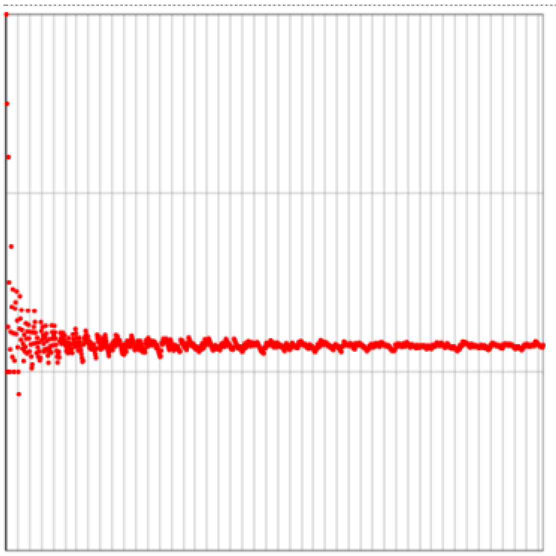
▪ Au niveau 1S : les élèves obtiennent une suite originale (Cette suite n'est ni arithmétique, ni géométrique, ni arithmético-géométrique. Cette suite n'est pas définie par récurrence et son terme général n'est pas donné en fonction de *n*). Cette particularité a, dans un premier temps, déstabilisé les élèves. Comme cette suite est croissante, cette situation offre l'occasion de traiter des problèmes de seuil en utilisant à nouveau un algorithme.

## *Documents*

### ▪ Algorithme donnant le nombre S de points en fonction de *n*

```
VARIABLES
  n EST_DU_TYPE NOMBRE
  k EST_DU_TYPE NOMBRE
  l EST_DU_TYPE NOMBRE
  compteur EST_DU_TYPE NOMBRE
  S EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
  LIRE n
  S PREND_LA_VALEUR 0
  POUR k ALLANT_DE 1 A sqrt(n)
    DEBUT_POUR
    POUR l ALLANT_DE 0 A sqrt(n)
      DEBUT_POUR
      SI (k*k+l*l<=n) ALORS
        DEBUT_SI
        compteur PREND_LA_VALEUR compteur+1
        FIN_SI
      FIN_POUR
    FIN_POUR
  S PREND_LA_VALEUR 4*compteur+1
AFFICHER S
FIN_ALGORITHME
```

### ▪ Graphique donnant une approximation de $\pi$ en fonction de *n*



Xmin: 0 ; Xmax: 1000 ; Ymin: 2 ; Ymax: 5 ; GradX: 22 ; GradY: 1

## ▪ Algorithme donnant une approximation de $\pi$ pour $n$=1000 (question 4)

```
VARIABLES
  n EST_DU_TYPE NOMBRE
  k EST_DU_TYPE NOMBRE
  l EST_DU_TYPE NOMBRE
  compteur EST_DU_TYPE NOMBRE
  S EST_DU_TYPE NOMBRE
  approxpi EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
  POUR n ALLANT_DE 1 A 1000
    DEBUT_POUR
    compteur PREND_LA_VALEUR 0
    S PREND_LA_VALEUR 0
    POUR k ALLANT_DE 1 A sqrt(n)
      DEBUT_POUR
      POUR l ALLANT_DE 0 A sqrt(n)
        DEBUT_POUR
        SI (k*k+l*l<=n) ALORS
          DEBUT_SI
          compteur PREND_LA_VALEUR compteur+1
          FIN_SI
        FIN_POUR
      FIN_POUR
    S PREND_LA_VALEUR 4*compteur+1
    approxpi PREND_LA_VALEUR S/n
    TRACER_POINT (n,approxpi)
    FIN_POUR
  AFFICHER approxpi

FIN_ALGORITHME
```

```
***Algorithme lancé***
3.149
```

## ▪ Algorithme donnant la plus petite valeur de $n$ telle que S>=A

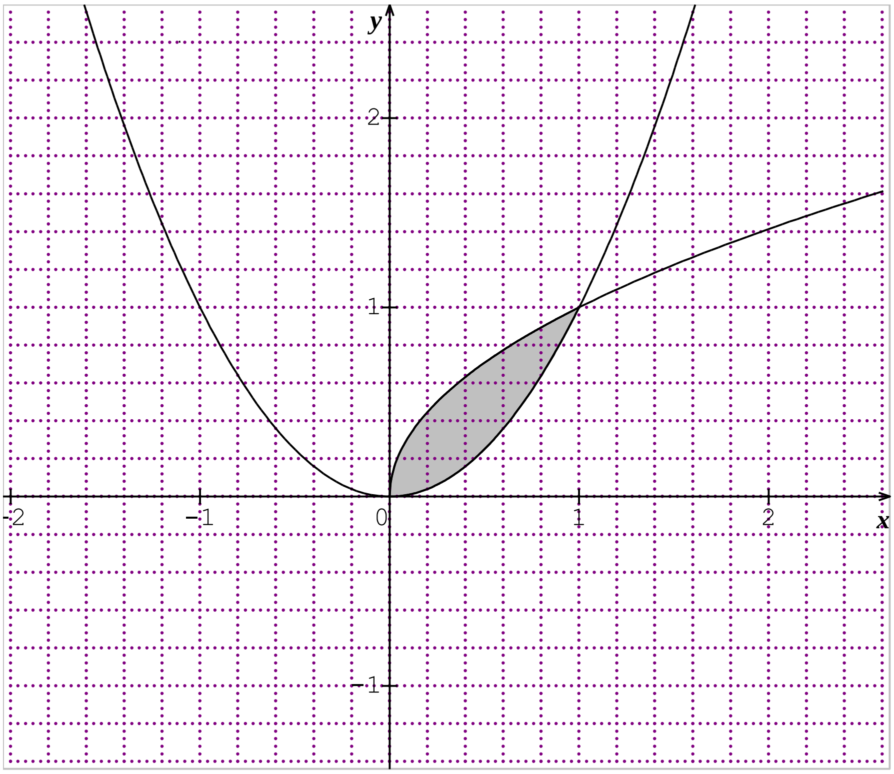
```
VARIABLES
  n EST_DU_TYPE NOMBRE
  k EST_DU_TYPE NOMBRE
  l EST_DU_TYPE NOMBRE
  compteur EST_DU_TYPE NOMBRE
  S EST_DU_TYPE NOMBRE
  A EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
  n PREND_LA_VALEUR 1
  LIRE A
  TANT_QUE (S<A) FAIRE
    DEBUT_TANT_QUE
    n PREND_LA_VALEUR n+1
    S PREND_LA_VALEUR 0
    compteur PREND_LA_VALEUR 0
    POUR k ALLANT_DE 1 A sqrt(n)
      DEBUT_POUR
      POUR l ALLANT_DE 0 A sqrt(n)
        DEBUT_POUR
        SI (k*k+l*l<=n) ALORS
          DEBUT_SI
          compteur PREND_LA_VALEUR compteur+1
          FIN_SI
        FIN_POUR
      FIN_POUR
    S PREND_LA_VALEUR 4*compteur+1
    FIN_TANT_QUE
  AFFICHER "Quand n >= "
  AFFICHER n
  AFFICHER " le nombre de points est supérieur ou égal à "
  AFFICHER A
FIN_ALGORITHME
```

RESULTATS :

```
***Algorithme lancé***
Quand n >= 320 le nombre de points est supérieur ou égal à  1000
```

**Annexe**
**Recherche donnée quelques semaines avant :**

Enoncé
Sur le graphique ci-dessous, on a colorié en gris le domaine compris entre la courbe représentative de la fonction carré et la courbe représentative de la fonction racine carrée.




Créer un algorithme qui indique si un point A donné par ses coordonnées appartient ou pas à la zone grise.

## L'ALGORITHME D'ELISE

```
VARIABLES
  xA EST_DU_TYPE NOMBRE
  yA EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
  LIRE xA
  LIRE yA
  AFFICHER "xA = "
  AFFICHER xA
  AFFICHER "yA = "
  AFFICHER yA
  SI (0<=xA ET xA<=1 ET pow(xA,2)<=yA et yA<=sqrt(xA)) ALORS
    DEBUT_SI
    AFFICHER "Le point A se situe dans la zone grise."
    FIN_SI
    SINON
      DEBUT_SINON
      AFFICHER "Le point A ne se situe pas dans la zone grise."
      FIN_SINON
FIN_ALGORITHME
```

## L'algorithme de Stephane

```
VARIABLES
  Xa EST_DU_TYPE NOMBRE
  Ya EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
  LIRE Xa
  LIRE Ya
  SI (Xa>=0 ET Xa<=1) ALORS
    DEBUT_SI
    SI (Ya<=sqrt(Xa) ET Ya>=Xa*Xa) ALORS
      DEBUT_SI
      AFFICHER "Le point A appartient à la zone grise."
      FIN_SI
      SINON
        DEBUT_SINON
        AFFICHER "Le point A n'appartient pas à la zone grise."
        FIN_SINON
    FIN_SI
    SINON
      DEBUT_SINON
      AFFICHER "Le point A n'appartient pas à la zone grise"
      FIN_SINON
FIN_ALGORITHME
```

## L'algorithme de Paul (le grand) :

```
VARIABLES
  xA EST_DU_TYPE NOMBRE
  yA EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
  LIRE xA
  LIRE yA
  SI (xA>=0) ALORS
    DEBUT_SI
    SI (yA>=(xA*xA) ET yA<=(sqrt(xA))) ALORS
      DEBUT_SI
      AFFICHER "Le point A se situe dans la zone zone grise"
      FIN_SI
      SINON
        DEBUT_SINON
        AFFICHER "Le point A ,ne se situe pas dans la zone grise"
        FIN_SINON
    FIN_SI
    SINON
      DEBUT_SINON
      AFFICHER "Le point A ne se situe pas dans la zone grise"
      FIN_SINON
FIN_ALGORITHME
```