

TP n°8

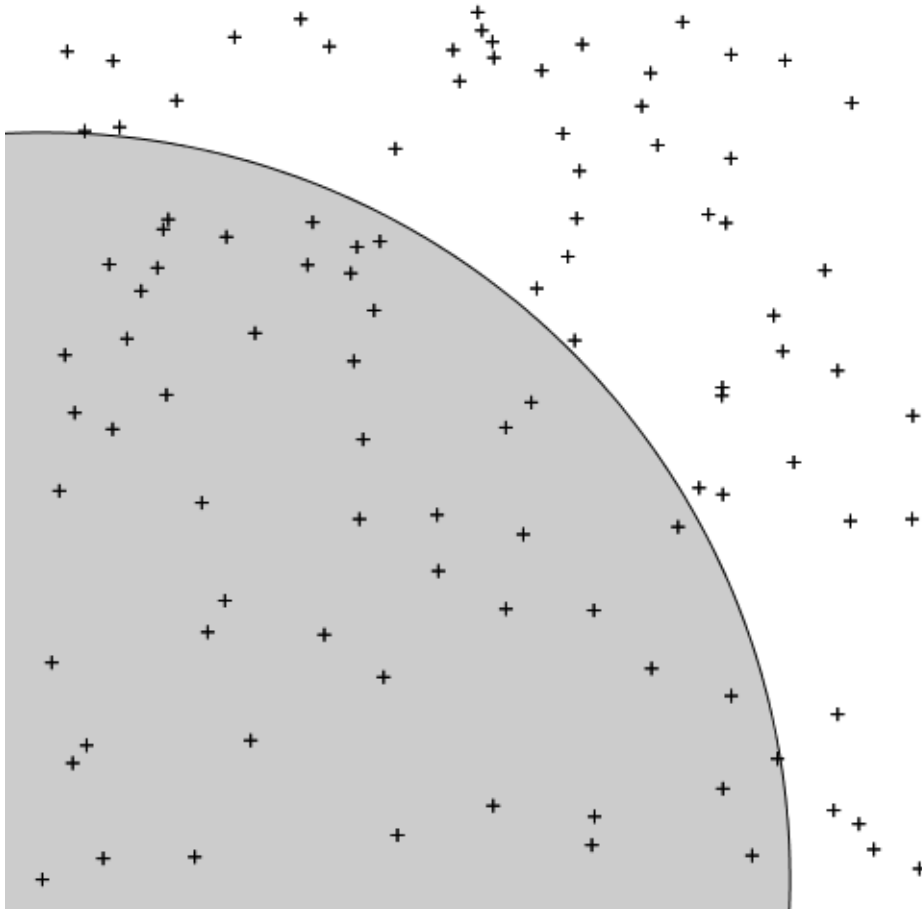
Problème d'englobement

Monsieur Buffalo a de la chance : Une centaine d'arums ont fleuri dans son pâturage de forme carrée, à des endroits aléatoires. Il souhaite délimiter par un enclos de forme circulaire, une zone comprenant la moitié de ces arums.

Comme dans le TP précédent, on va construire un nuage de 100 points dans le carré de côté 1 par

```
for(i=1 ; i<=100 ; i=i+1){  
x=Math.random() ;  
y=Math.random() ;  
p=Point(x,y) ; }
```

Seulement cette fois-ci, on demande de construire un cercle encerclant exactement la moitié des points du nuage, c'est-à-dire qu'il doit y avoir 50 points à l'intérieur du cercle et 50 points à l'extérieur de celui-ci :



Comme on peut construire en JavaScript un cercle de centre (0,0) et de rayon donné (avec "c=FixedCircle(o,r)"), il suffit de trouver le rayon à donner à ce cercle (après avoir créé le point "o" dont les deux coordonnées sont nulles).

Il est possible de créer 100 variables (pour les distances) d'un coup, avec

```
var distances=new Array() ;
```

Alors la première des variables s'appellera `distances[1]`, la seconde variable s'appellera `distances[2]`, ... et la centième s'appellera `distances[100]` (en fait JavaScript considère que "distances" est une seule variable, de type tableau - ce qui se dit *array* en Anglais).

L'intérêt de ce "tableau" de nombres est qu'on peut y accéder automatiquement par `distances[i]`, où `i` est la valeur de l'indice dans la boucle.

Par ailleurs, il est possible de trier automatiquement dans l'ordre croissant les nombres qui sont dans le tableau `distances` en JavaScript en entrant

```
distances=distances.sort() ;
```

(en Anglais, trier se dit "to sort")

Question subsidiaire

Ajouter à la construction suivante deux autres cercles, l'un contenant 25 points, l'autre contenant 75 points (donc laissant 25 points en-dehors de lui) :

