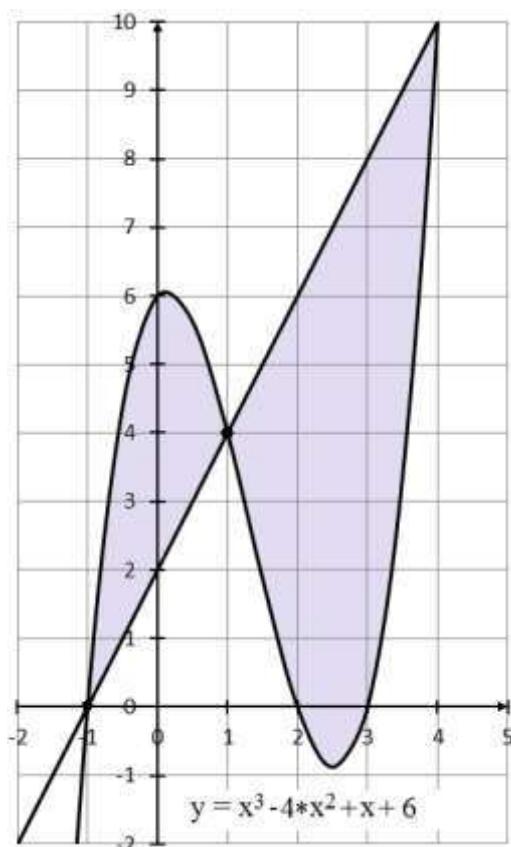


## Задание 2

### Постановка задачи

Написать программу, которая определяет, попадает ли точка с заданными координатами в заштрихованную область. Точки на границе принадлежат области. Необходимые параметры получить из рисунка. Результат работы программы вывести в виде текстового сообщения: Попадает, Не попадает.



### Теоретическое введение

Для решения задачи воспользуемся условным оператором:

**if** <Логическое выражение>:

    <Блок – выполняется, если условие истинно>

[**elif** <Логическое выражение>:

    <Блок – выполняется, если условие истинно>

]

[**else**:

    <Блок – выполняется, если все условия ложны>

]

<Блок> – это набор инструкций, которые выделяются одинаковым количеством пробелов (обычно четырьмя).

Необходимо правильно составить логическое выражение, параметрами которого будут значение координат точки ( $x, y$ ) и уравнения линий.

Обмен с консолью выполняется стандартными функциями ввода/вывода: `input()` и `print()`.

Для решения задачи требуется знать уравнение прямой (уравнение кривой приведено на рисунке). Из рисунка получаем координаты двух точек, через которые проходит прямая линия  $(-1, 0)$  и  $(1, 4)$ . Подставив значения выбранных точек в уравнение общего вида  $y = kx + b$ , получим систему уравнений. Решив эту систему, найдём значения для параметров  $k$  и  $b$ . В итоге уравнение прямой примет вид:  $y = 2x + 2$ .

Точка с координатами  $(x, y)$ , введённая пользователем, будет попадать в заштрихованную область на интервале  $[-1, 1]$  в том случае, если  $x$  будет не меньше  $-1$  и не больше  $1$ . Кроме этого,  $y$  должен быть не меньше значения полученного для прямой в точке  $x$  и не больше значения, вычисленного для кубической параболы в этой точке.

### Описание алгоритма

1. Ввести координаты точки  $(x, y)$  и привести значения к типу `float`.
2. Выполнить проверку на попадание точки в заданную область.
3. Вывести результат в виде: "Точка  $x, y$  попадает в область." и "Точка  $x, y$  не попадает в область."

### Описание входных и выходных данных

Входные данные - координаты точки, введённые пользователем. Тип данных и точность представления в задаче не заданы. Установим вещественный тип (`float`).

Выходные данные - сообщения, в текстовом виде, о попадании или непопадании точки в заданную область.

### Тестовые примеры

X	Y	Результат
-1	0	Попадает
-0.5	-1	Не попадает
0	3	Попадает
1	4	Попадает
1	5	Не попадает
1.5	1	Не попадает
2	3	Попадает
2.5	-1	Не попадает
2.5	-0.3	Попадает
3.5	10	Не попадает

### Листинг программы

```
from math import *
flag = 0
print('Введите координаты X и Y для точки:')
x = float(input('X='))
```

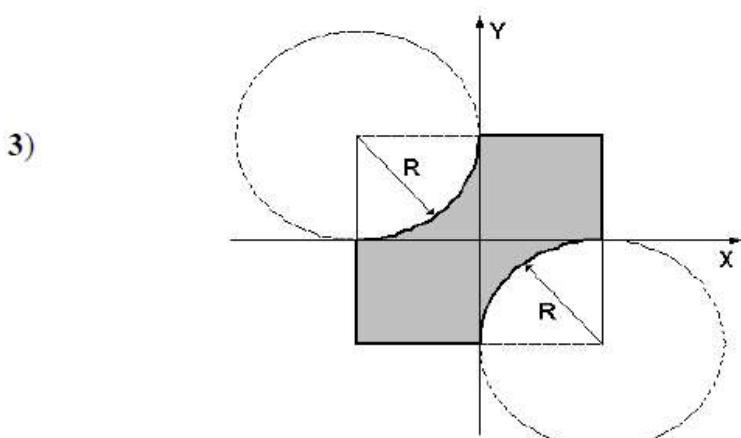
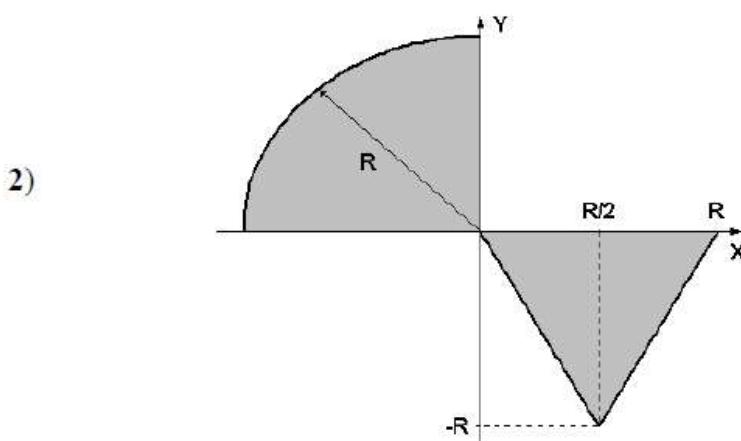
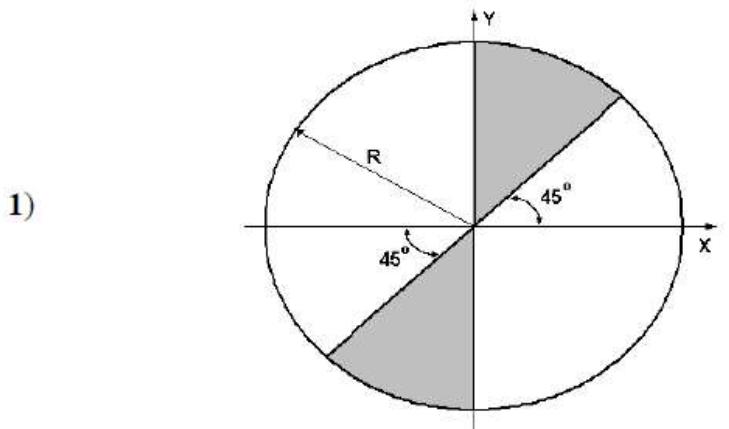
```
y = float(input('Y='))  
if (x < -1) or (x > 4):  
    flag = 0      #False  
if ((x>=-1) and (x<1) and (y>=2*x+2)  
    and (y<=x**3-4*x**2+x+6) or (x>=1)  
    and (x<=4) and (y>=x**3-4*x**2+x+6) and (y<=2*x+2)):  
    flag = 1  
else:  
    flag = 0  
print("Точка X={0: 6.2f} Y={1: 6.2f}"  
      .format(x, y), end=" ")  
if flag:  
    print("попадает", end=" ")  
else:  
    print("не попадает", end=" ")  
print("в область.")
```

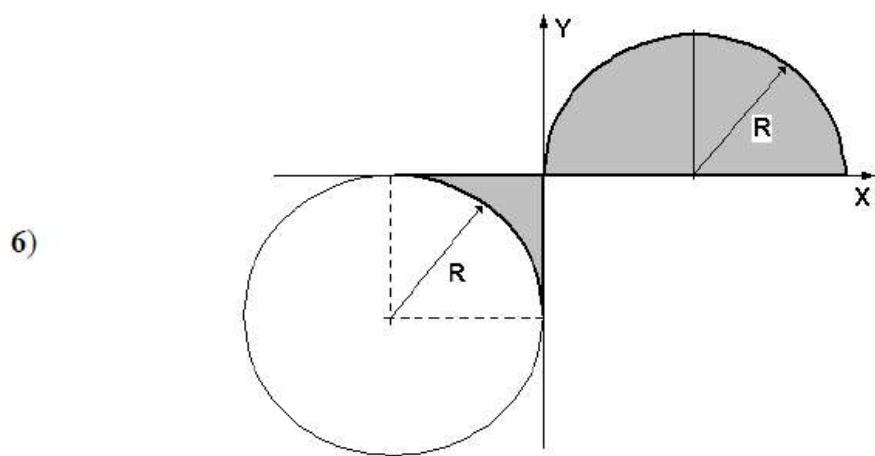
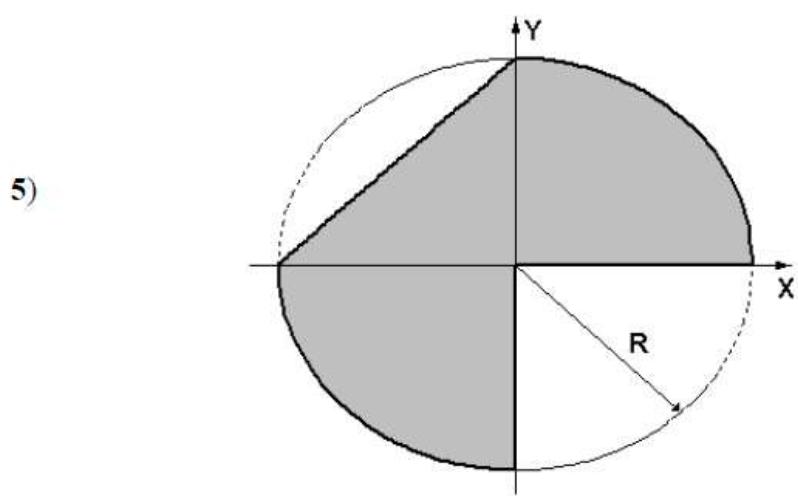
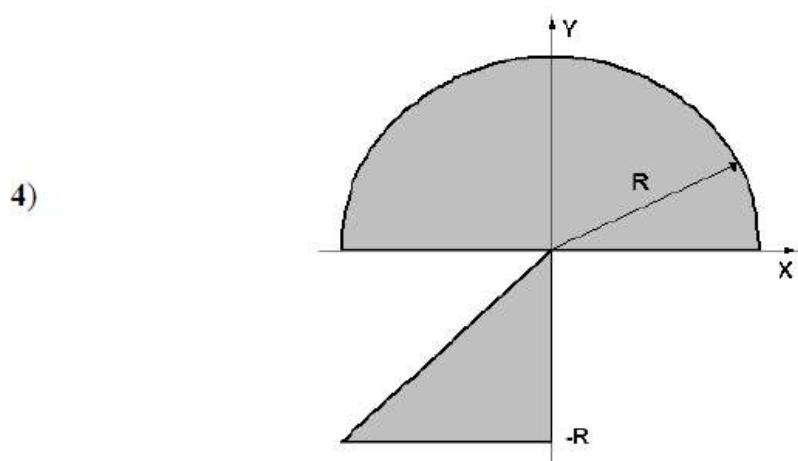
## Задание к лабораторной работе №2

### «Разветвляющиеся вычислительные процессы».

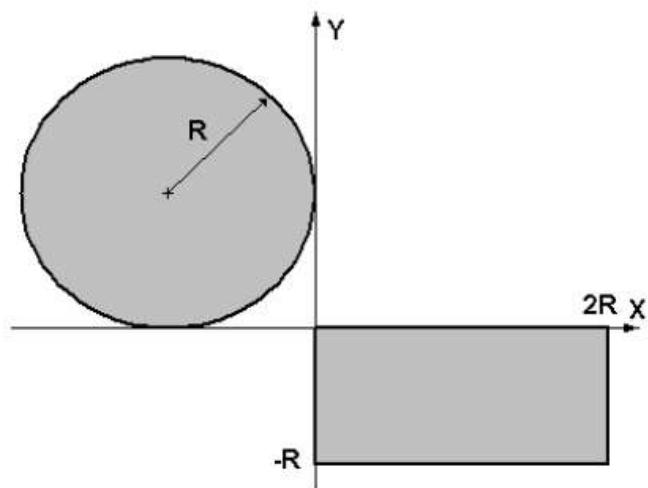
#### Задание 2

Написать программу, которая определяет, попадает ли точка с заданными координатами  $X$ ,  $Y$  в область, закрашенную на рисунке серым цветом. Результат работы программы вывести в виде текстового сообщения. Параметр  $R$  вводится с клавиатуры.

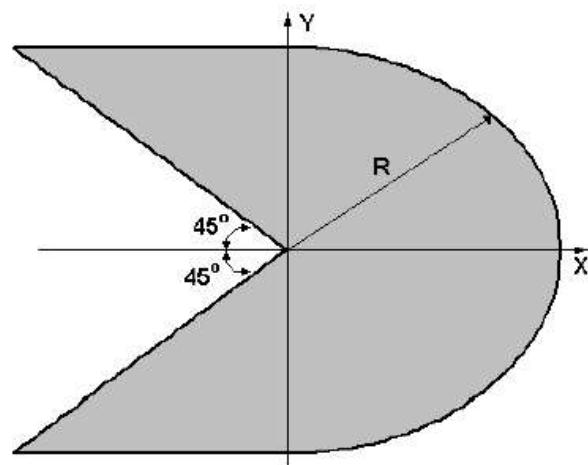




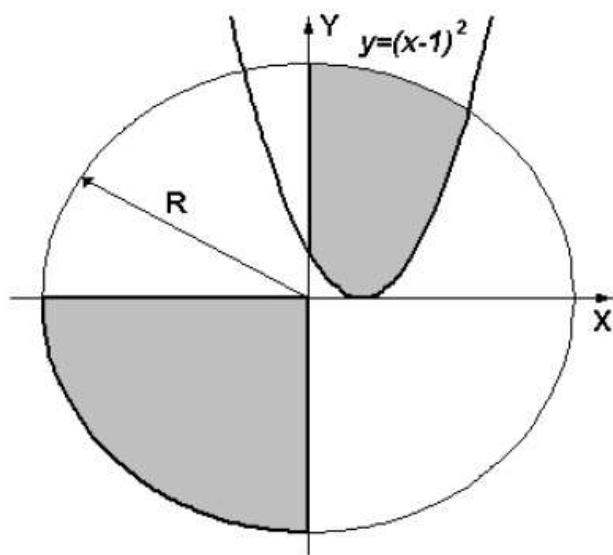
7)



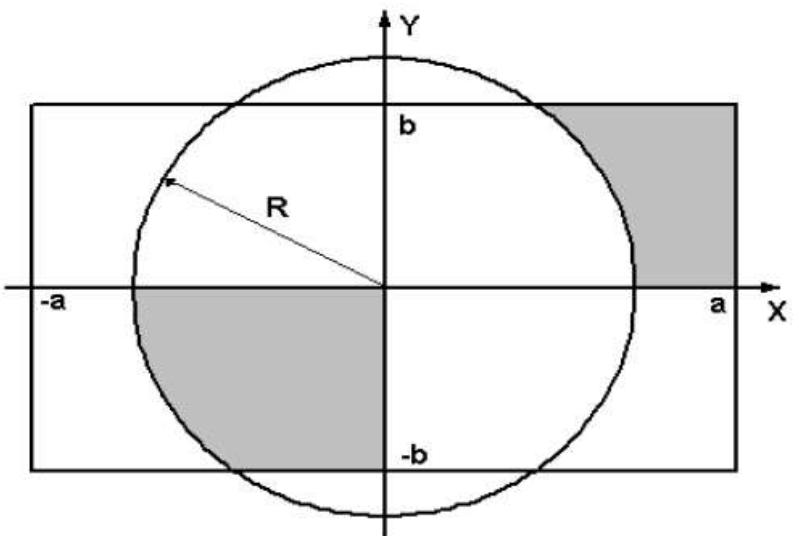
8)



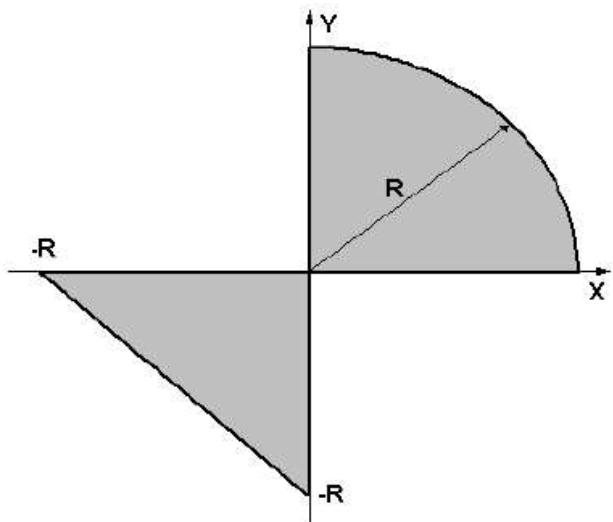
9)



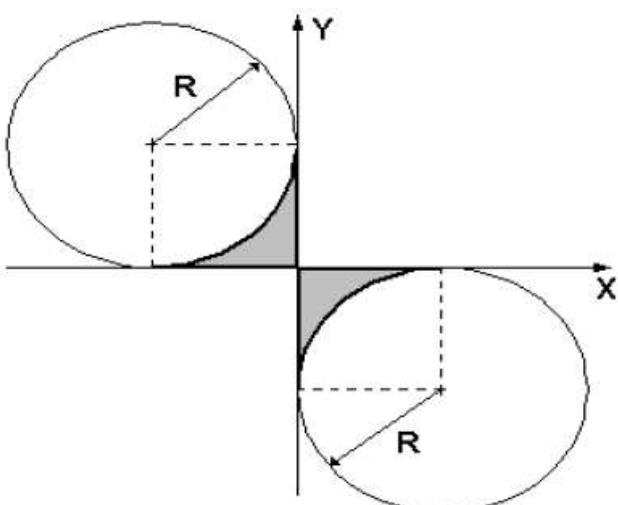
10)

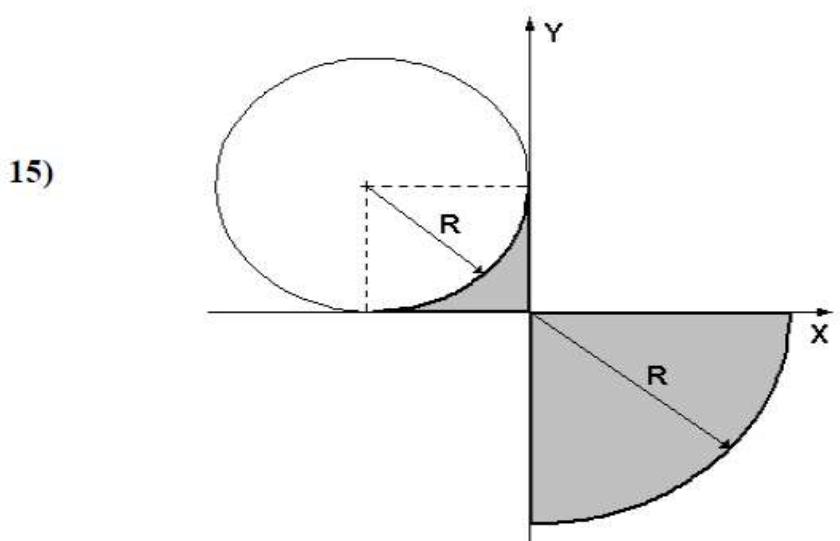
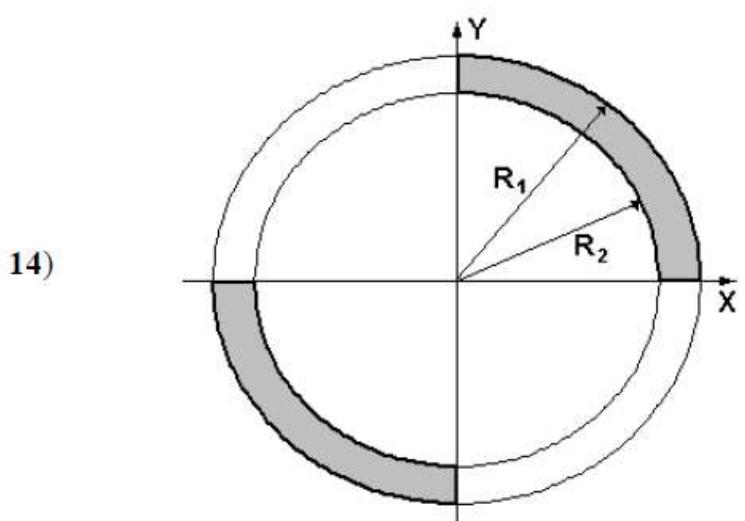
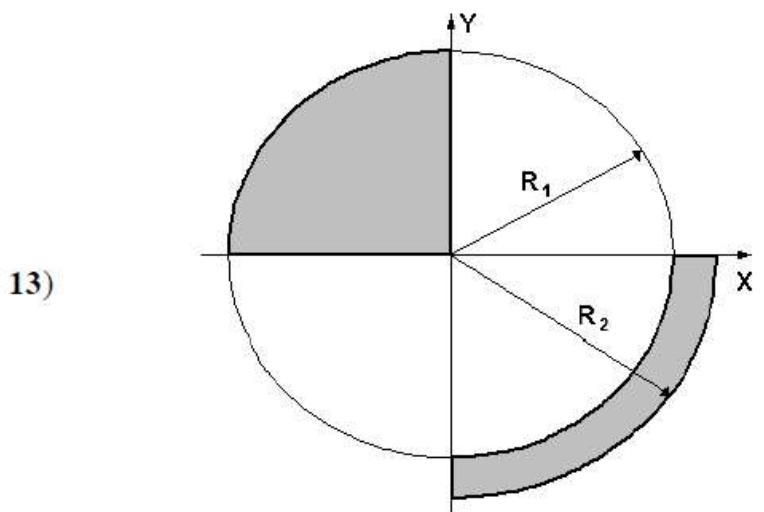


11)

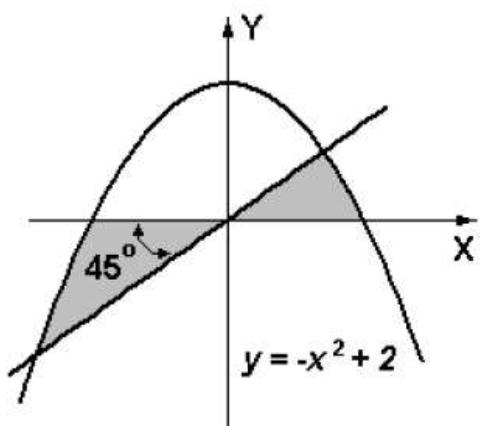


12)

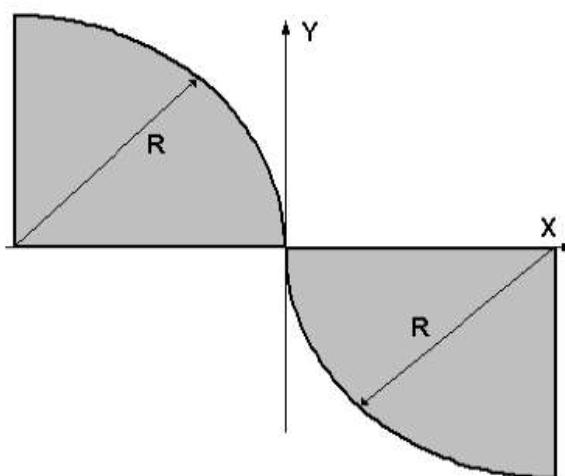




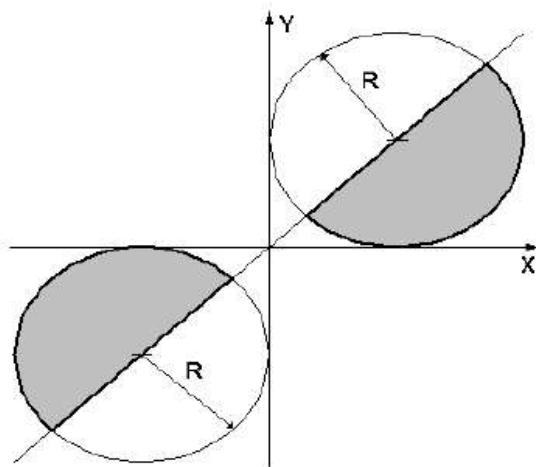
16)

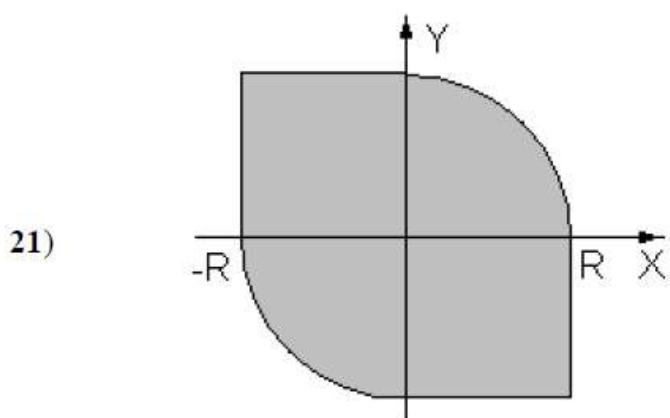
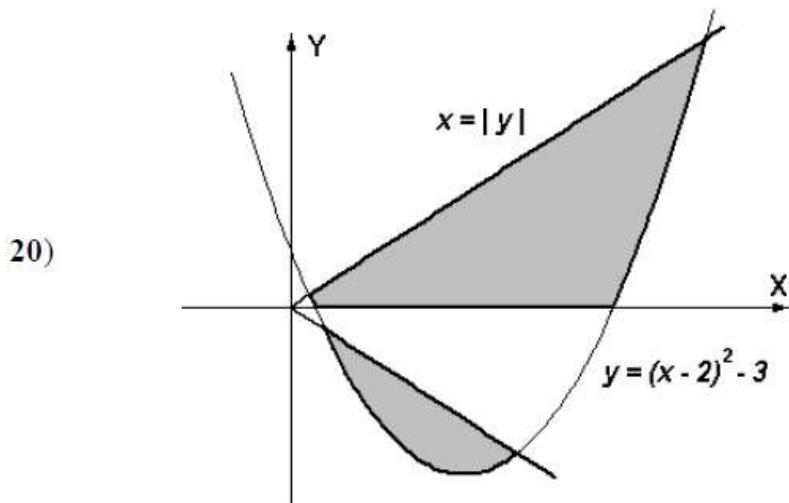
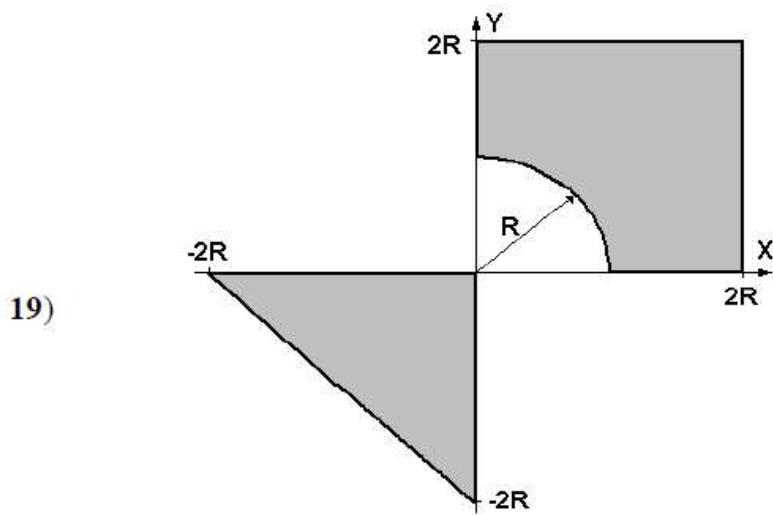


17)

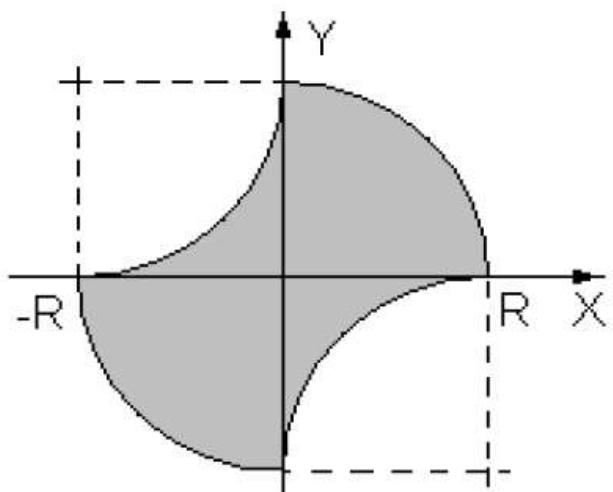


18)

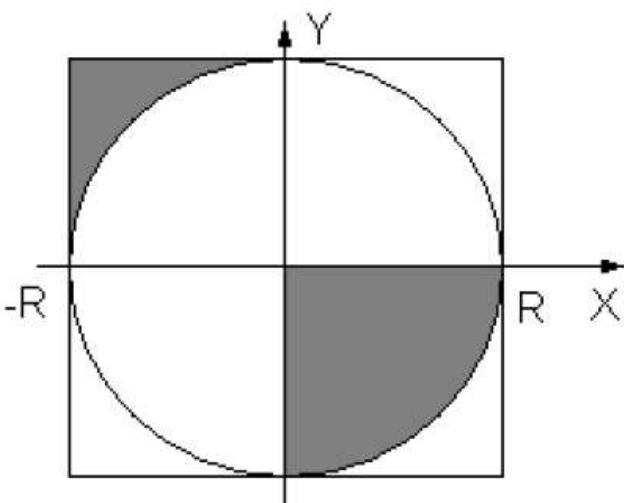




22)



23)



24)

