

Объектно-ориентированное программирование

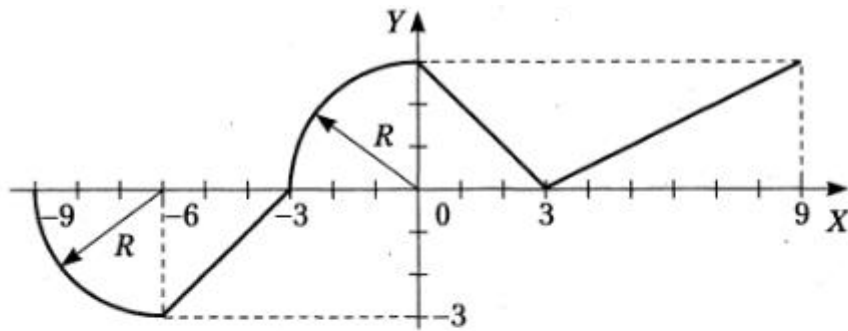
Домашнее задание №2

Задание:

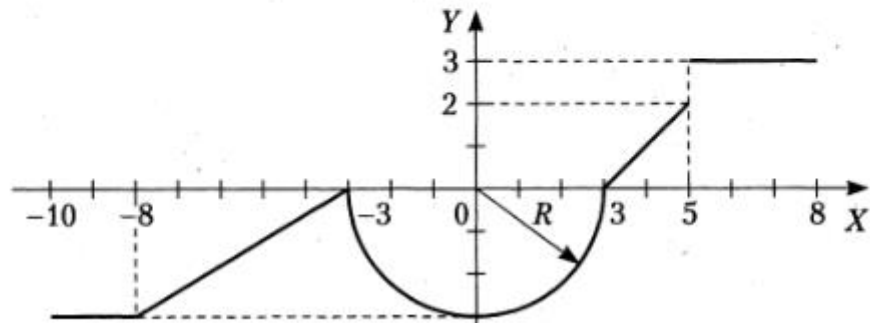
Необходимо написать программу, которая по введенному значению аргумента вычисляет значение функции, заданной в виде графика. Параметр R вводится с клавиатуры. В случае если значение R меньше указанного на графике, то в функции появляются разрывы, на которых функция не определена.

Варианты:

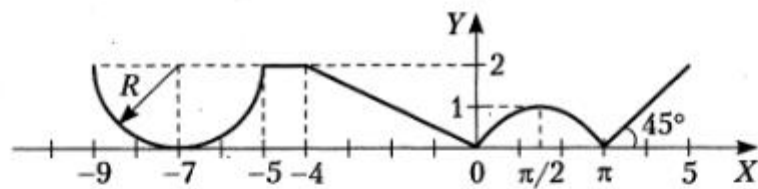
1



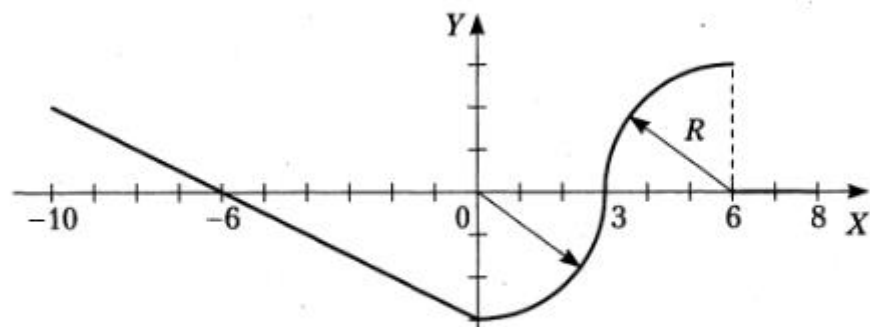
2



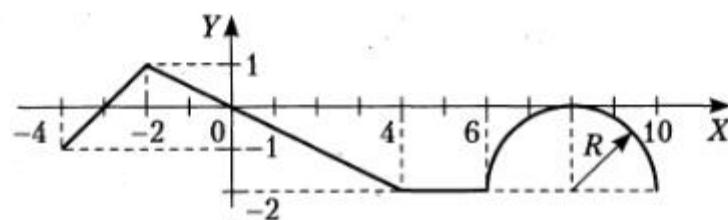
3



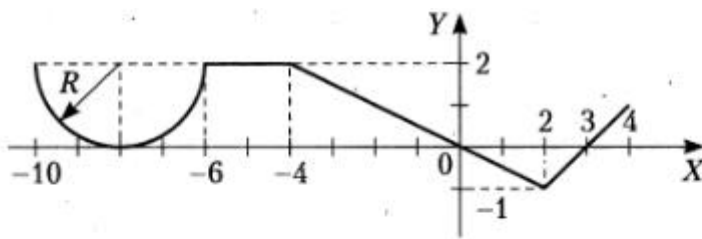
4



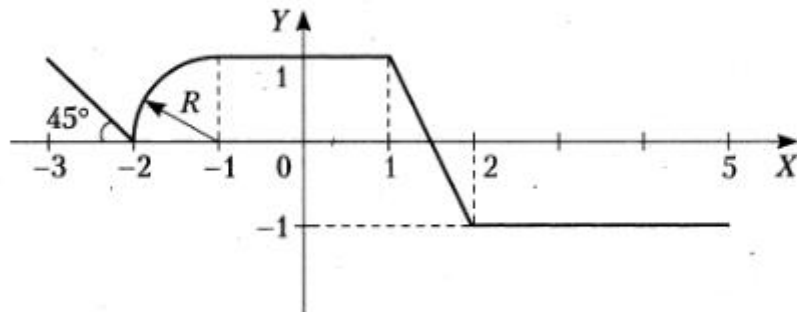
5



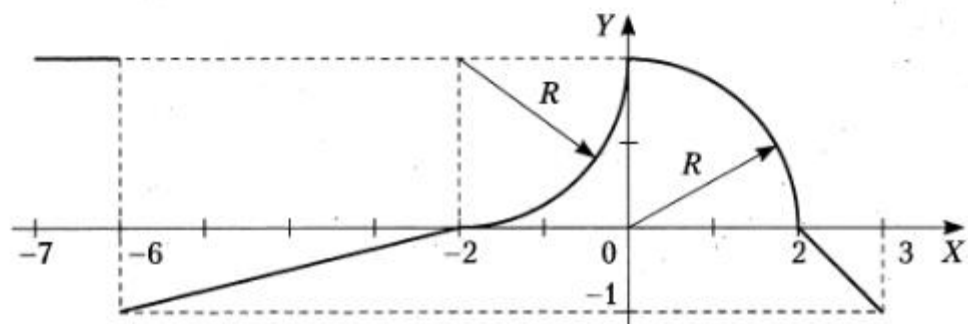
10



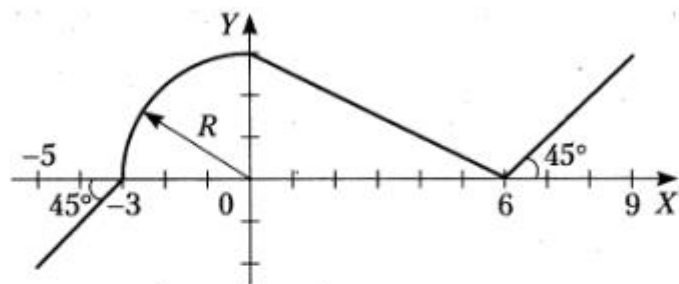
11



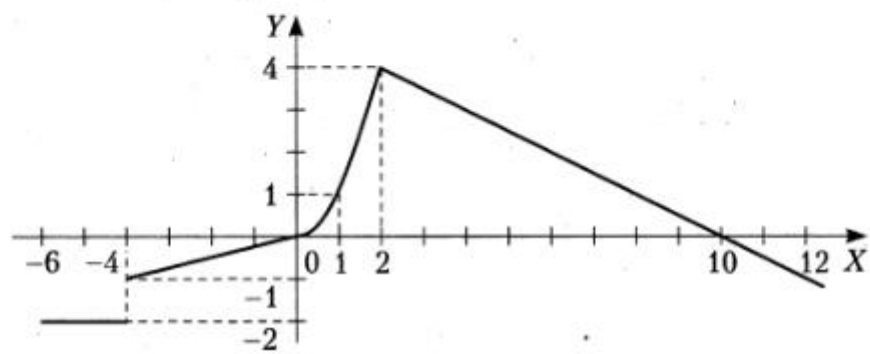
12



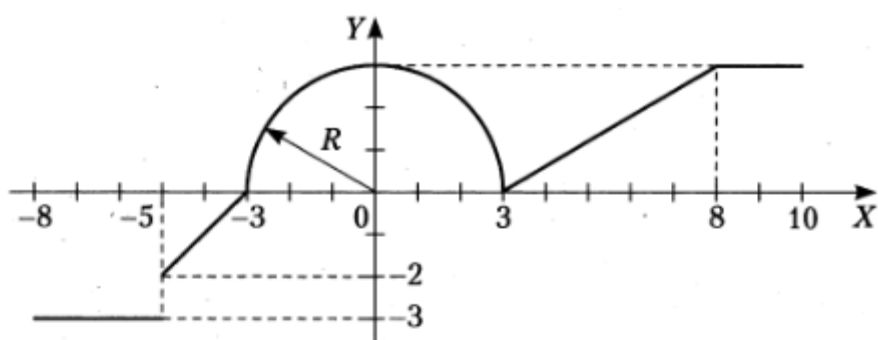
13



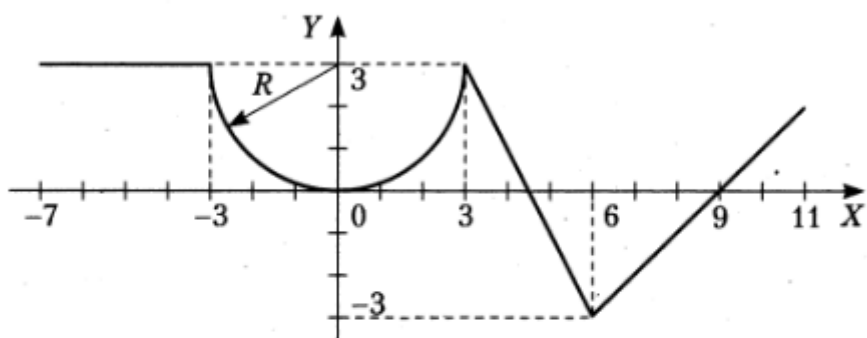
14



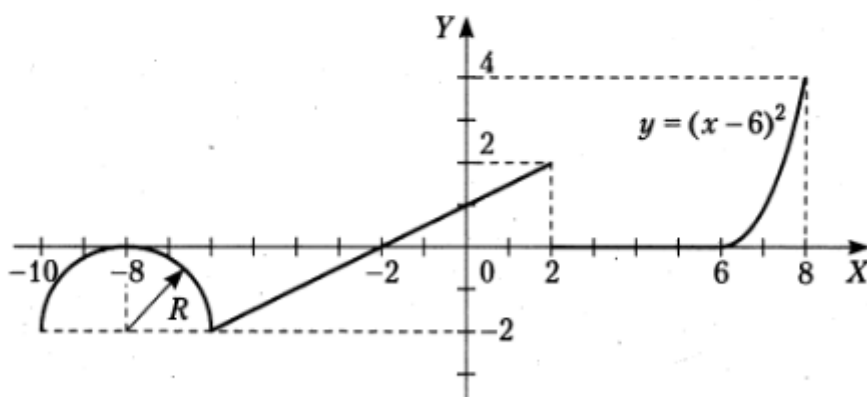
6



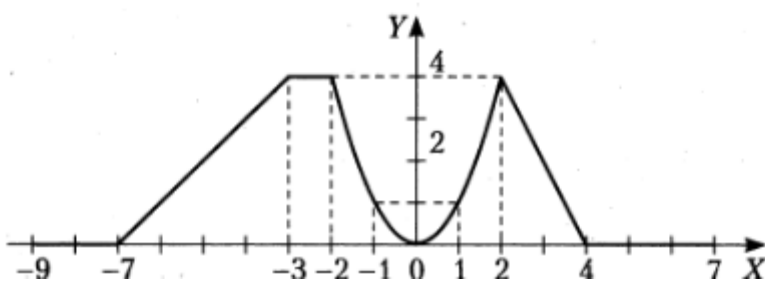
7



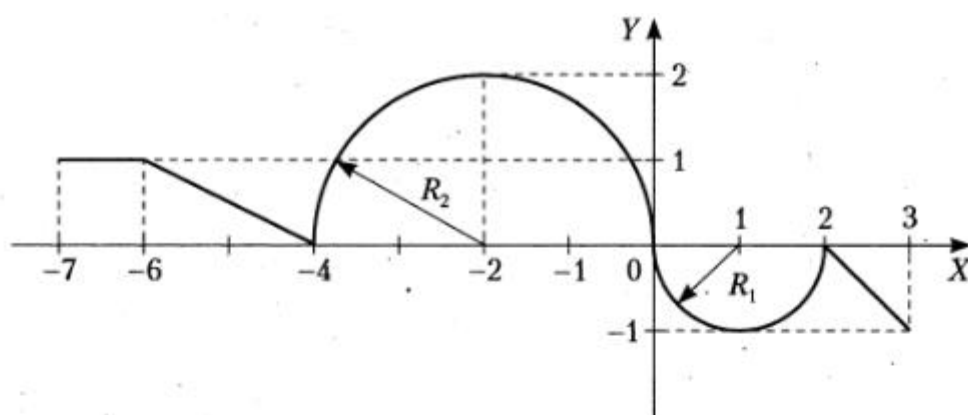
8



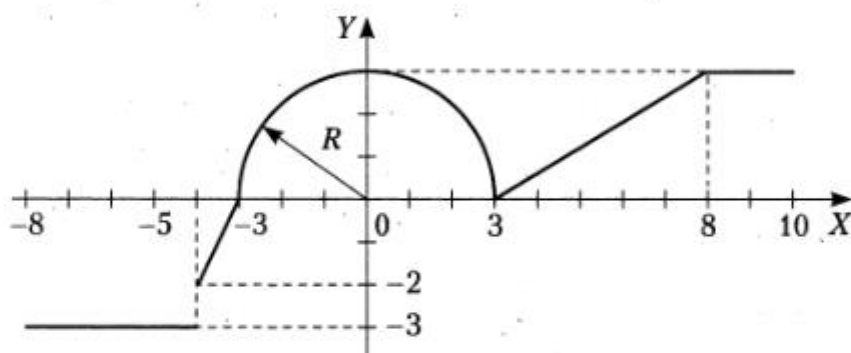
9



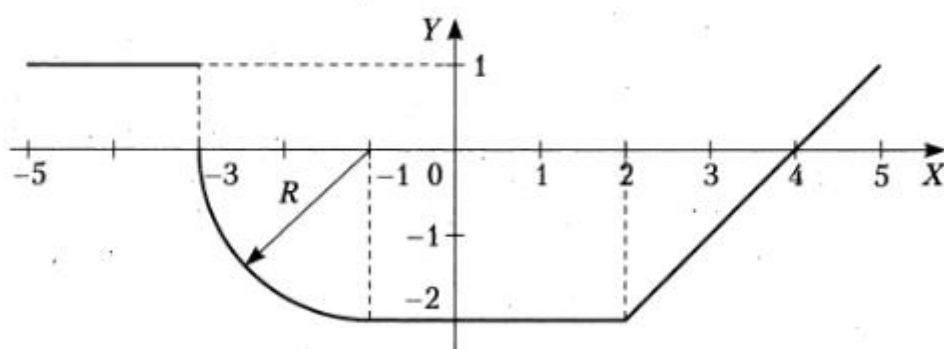
15



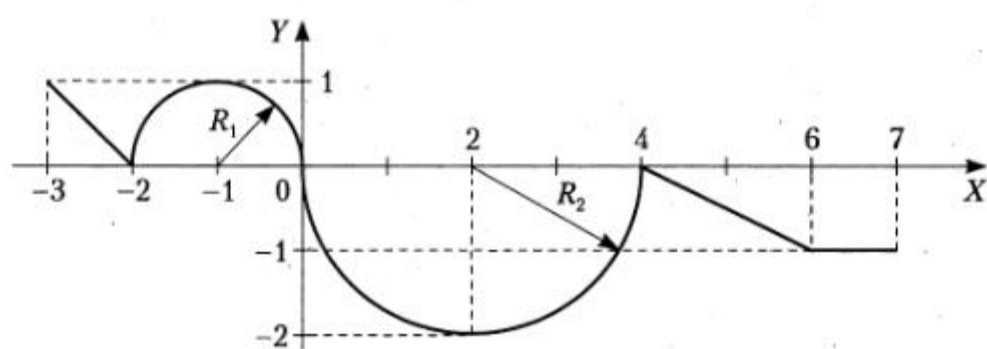
16



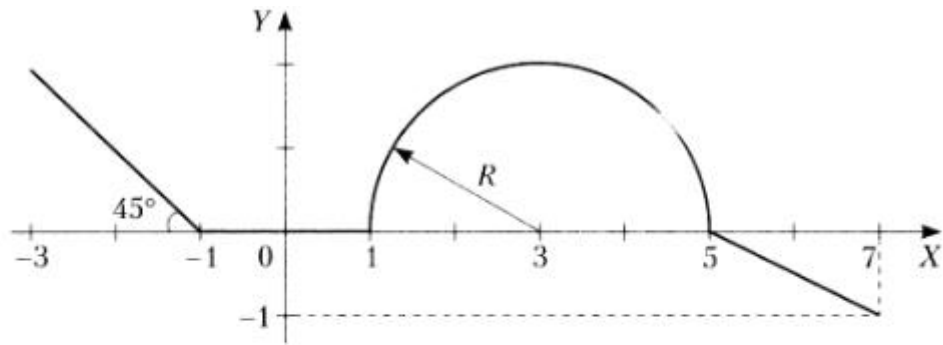
17



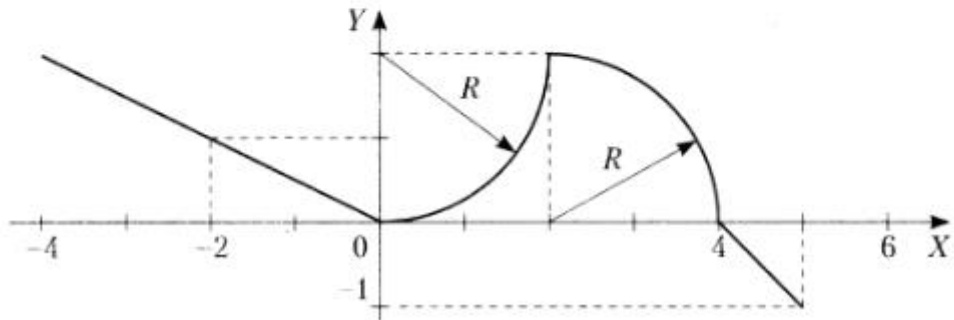
18



19



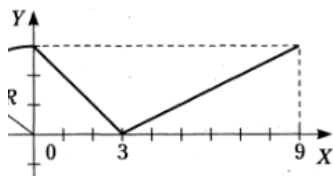
20



Ход решения

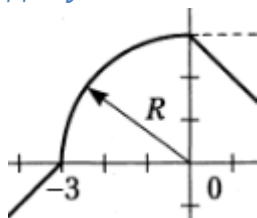
1. Разбить график на набор известных функций.

Для начала необходимо разбить график на составные участки, и вывести зависимость значения функции от значения аргумента. Например, если функция на участке определена прямой, например:



Очевидно, что она описывается уравнением вида $y(x) = kx + b$. В данном случае для первого участка $y(x) = -x + 3$; а для второго $y(x) = 0.5(x - 3)$.

Для участка заданного окружностью:

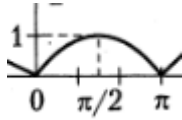


функцию можно вывести различными способами: проекцией векторов, уравнением окружности или через прямоугольный треугольник. В данном примере: $y(x) = -R \sin \arccos x/R$, либо $y(x) = \sqrt{R^2 - x^2}$. Если окружность находится не в начале координат, то это нужно будет учесть в расчетах.

При вычислении значения функции, согласно заданию, нужно позволить пользователю указывать собственное значение радиуса окружностей. Если пользователь задаст значение радиуса

меньшее, чем отображено на графике, то функция будет иметь область неопределенных значений, при выводе таблицы значений об этом нужно будет предупредить пользователя соответствующим сообщением.

Для участка заданного (ко)синусоидой:



Функция – это просто (ко)синус от аргумента: $y(x) = \sin x$ или $y(x) = \cos x$.

2. Создать методы для расчета значений на отрезках

После того, как график разделен на отрезки, и определены функции которыми они заданы, необходимо для каждого отрезка создать метод для расчета значения на каждом из отрезков. Вы можете назвать эти методы `segment1`, `segment2` и тд. Например:

```
//расчет первого сегмента графика
static double segment1(double x, double r){
    //центр окружности
    double a = -6;
    double b = 0;
    double y;
    //по уравнению окружности
    y = Math.Sqrt(r*r - (x-a)*(x-a)) - b;

    //возвращаем значение в нижней четверти
    return -y;
}
```

Вы должны определить, какие параметры будет принимать такой метод, и какие данные будет возвращать.

Написав методы для каждого отрезка, вы можете проверить верность расчетов, просто подставив пару значений и сверив с графиком.

3. Распечатать таблицу значений функции

После того, как все методы для расчета отрезков написаны, необходимо собрать их в одно целое. Для этого можно воспользоваться оператором условия (if):

```
if (x < -9) {
    Console.WriteLine ("Функция не определена");
} else if (x < -6) {
    Console.WriteLine ("y({0:0.00}) = {1:0.00}", x, segment1 (x, r));
} else if (x < -3) {
    Console.WriteLine ("y({0:0.00}) = {1:0.00}", x, segment2 (x, r));
}
```

Таким образом для каждого отрезка будут вызваны соответствующие ему методы.

И, наконец, для того чтобы можно было напечатать таблицу значений, нужно воспользоваться циклом for, для перебора всех возможных значений X (с шагом 0.2):

```
for (double x = -9; x <= 9; x += 0.2) {  
  
    if (x < -9) {  
        Console.WriteLine ("Функция не определена");  
    } else if (x < -6) {  
        Console.WriteLine ("y({0:0.00}) = {1:0.00}", x, segment1 (x, 3));  
    } else if (x < -3) {  
        Console.WriteLine ("y({0:0.00}) = {1:0.00}", x, segment2 (x, 3));  
    }  
    .....  
}
```

После этого необходимо добавить запрос у пользователя значения R для окружностей, и учесть его в расчетах.