

Лабораторная работа №1

По дисциплине: «Основы алгоритмизации и программирования»

Тема работы: «Программирование разветвляющихся процессов»

Для специальности: «Автоматизированные системы обработки информации и управления» (2202)

Составитель: преподаватель специальных дисциплин
Клопов Д.А.

Рассмотрено и одобрено на заседании
цикловой методической комиссии
спецдисциплин 2202

Протокол № _____ от «___» _____ 2006 г.

Председатель ЦМК _____ / Д.А. Клопов /

Цель работы:

Изучение и практическое применение управляющих операторов языка Turbo Pascal для организации разветвляющихся процессов.

Задание:

Используя имеющиеся в наличии задания, разработать в соответствии со своим вариантом программу, позволяющую определить по введенным с клавиатуры координатам точки:

для вариантов 1-20: принадлежит ли данная точка графику функции.

для вариантов 20-35: принадлежит ли данная точка «заштрихованной» области.

Этапы выполнения

Этап 1. Начало работы

Загрузите Turbo Pascal и создайте новый файл, используя команду **“File-New”**. Будет открыто окно редактора с именем **NONAME00.PAS**. Затем сохраните файл, используя команду **“File-Save as...”** в своем каталоге под именем **“OAIP_L01.PAS”**. Если Вы уже выполнили часть лабораторной работы ранее, и продолжаете работу над ней, то вместо команды **“File-New”** используйте команду **“File-Open” (F3)** для загрузки созданного ранее файла. Более подробно операции по работе в среде Turbo Pascal рассмотрены в [3].

Этап 2. Программирование и отладка

Составьте блок-схему для решения задачи. При составлении блок-схемы руководствуйтесь правилами по ее оформлению [2].

Далее, по составленной блок-схеме, составьте исходный текст программы. Внесите его в окно редактора Turbo Pascal и сохраните файл.

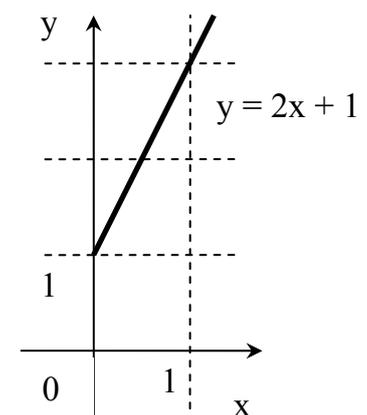
Добейтесь работоспособности программы и проверьте правильность вычислений. Для проверки правильности вычислений составьте контрольный пример и проведите расчеты «вручную».

Пример решения задачи

Условие задачи: По введенным с клавиатуры координатам точки определить, принадлежит ли она графику функции (рис.1)

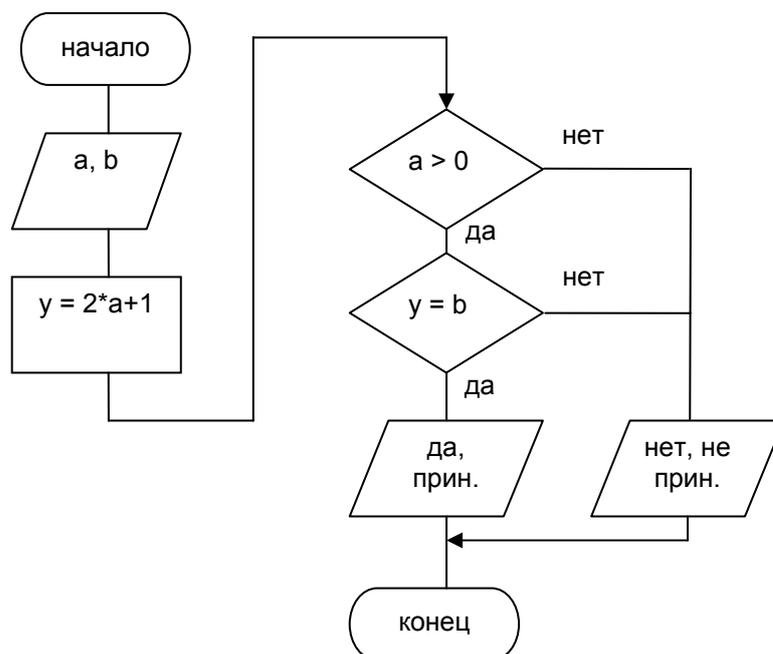
Из условия задачи видно, что функция является «лучом» с началом в точке с координатами (0;1). Следовательно, при составлении программы необходимо ограничить область значений аргумента интервалом от 0 до $+\infty$.

Обобщенный алгоритм решения задачи выглядит следующим образом:



1. Вводим с клавиатуры координаты a (по X) и b (по Y) для некоторой точки C .
2. Рассчитываем значение функции $Y=2*a + 1$
3. Сравниваем полученное значение Y с введенным с клавиатуры значением b с учетом «попадания» значение a в область допустимых значений от 0 до $+\infty$.
4. Если значения Y и b совпали, значит делаем вывод о том, что точка $C(a;b)$ принадлежит графику функции $y=2x+1$. В противном случае – делаем обратный вывод.
5. Выводим на экран результат

Блок-Схема



Текст программы

```
Program OAIP_L01;
Var
  a,b : Real; {координаты точки}
  y    : Real; {расчетное значение функции}
Begin
  writeln('Введите координаты точки');      {Выводим пояснение}
  write('по X ='); readln(a);                {Вводим a}
  write('по Y ='); readln(b);                {Вводим b}

  y:=2*a+1;                                  {Рассчитываем значение Y}

  if (a > 0) and (y = b)                      {Проверяем, попадает ли значение }
    then writeln('да, принадлежит')           {в интервал от 0 до +∞ }
    else writeln('нет, не принадлежит')       {и совпадают ли }
                                               {расчетное и введенное значения функции}
End.
```

Замечание: Вместо одного сложного условия допустимо использование двух простых

Рекомендации по программированию

- Для организации «ветвления» алгоритма используйте управляющую конструкцию (оператор)
IF <условие> THEN <оператор1> ELSE <оператор2>;
или
IF <условие> THEN <оператор>;

- Координаты точки, для которой выполняются вычисления, определите как числовые переменные типа **INTEGER** или **REAL**
- Значения всех переменных, влияющих на результаты вычислений, следует вводить с клавиатуры
- Перед вводом значений с клавиатуры (оператор **READ** или **READLN**) следует вывести на экран сообщение (оператор **WRITE** или **WRITELN**), поясняющее пользователю текущие действия
- Результат вычислений следует вывести на экран

Примечание: Следует помнить, что если под «оператором» подразумевается несколько действий, то следует использовать операторные скобки **BEGIN-END**.

Рекомендации по отладке

- В качестве значений переменных, вводимых с клавиатуры, используйте значения контрольного примера
- Просмотреть результаты работы (переключиться на рабочий экран) можно при помощи команды “*Debug-User screen*” (**ALT-F5**). Возврат в редактор Turbo Pascal осуществляется нажатием любой клавиши. Более подробно операции по работе в среде Turbo Pascal рассмотрены в [3].

Составление отчета

Оформление отчета

Отчет о выполнении лабораторной работы оформляется в соответствии с методическими указаниями [2].

Содержание отчета

Отчет о выполнении лабораторной работы должен включать:

1. Титульный лист (пример оформления титульного листа см. в методических указаниях [2])
2. Цель работы
3. Задание в соответствии с вариантом
4. Блок-схему программы
5. Текст программы на языке Pascal
6. Результаты работы программы

Литература

1. Фаронов В.В. / Turbo Pascal 7.0. Начальный курс. Учебное пособие. М.: ОМД Групп, 2003
2. Клопов Д.А. / Методические указания «Оформление отчетов к лабораторным работам по дисциплине «Основы алгоритмизации и программирования».
3. Клопов Д.А. / Методические указания «Основы работы в среде Turbo Pascal 7.0».

Варианты заданий

Варианты с 1 по 20

Вариант	Задание
1.	
2.	
3.	
4.	

Вариант	Задание
5.	
6.	
7.	
8.	

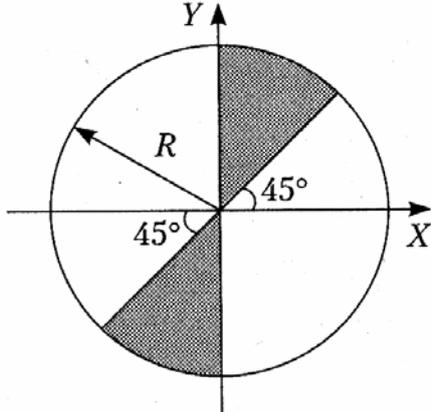
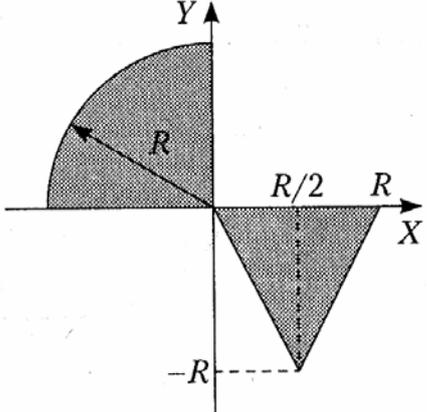
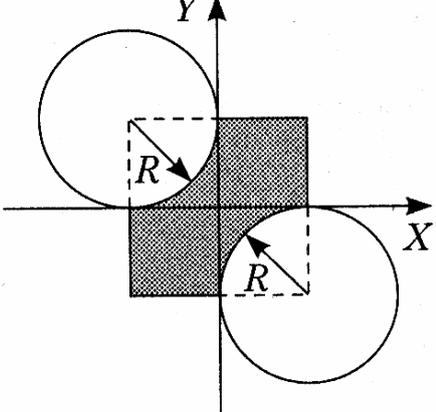
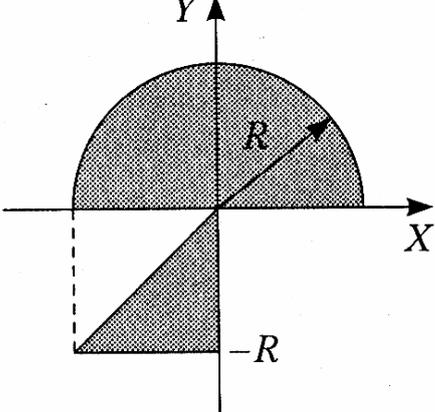
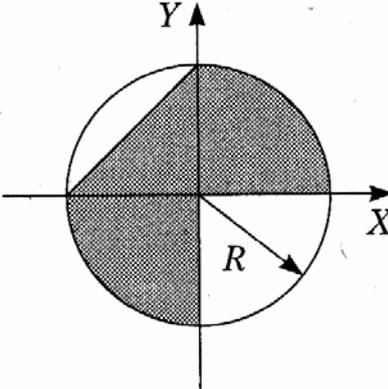
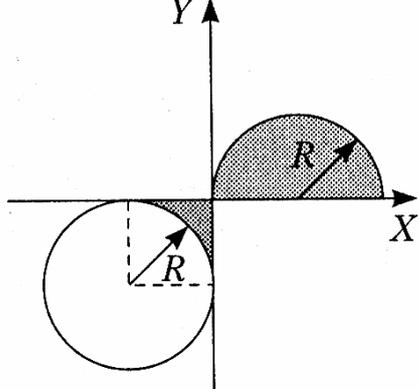
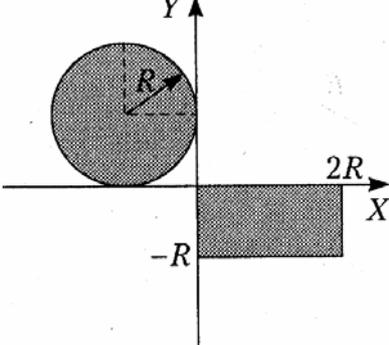
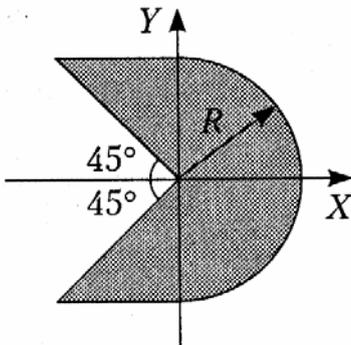
Вариант	Задание
9.	
10.	
11.	
12.	

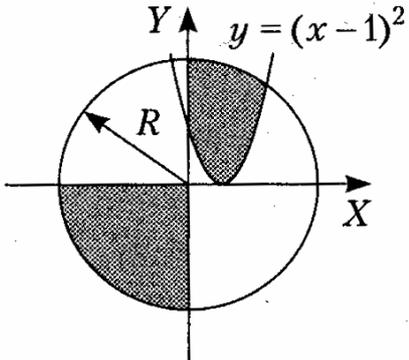
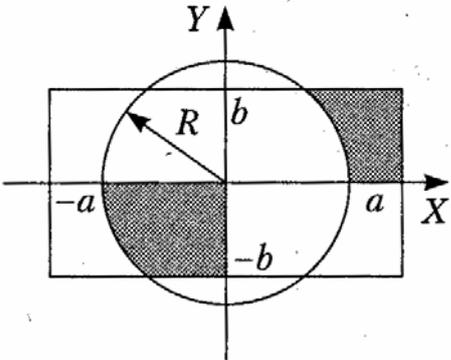
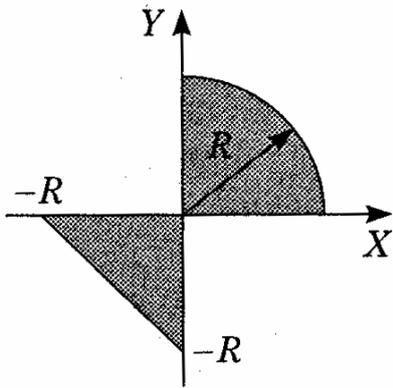
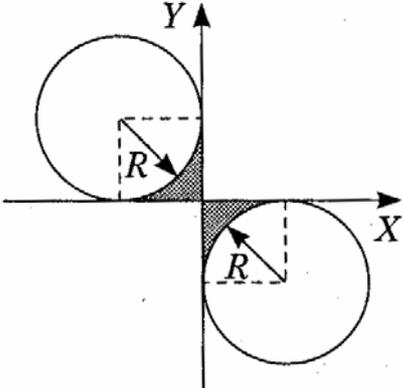
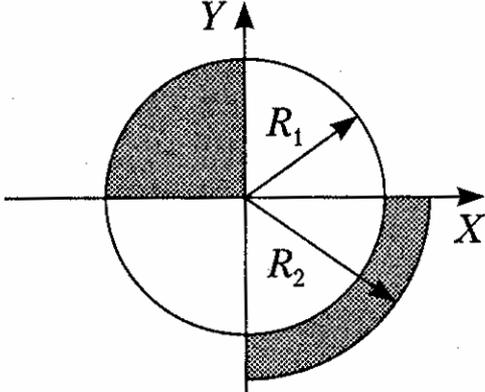
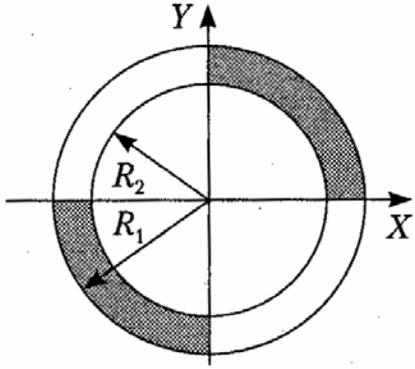
Вариант	Задание
13.	
14.	
15.	
16.	

Вариант	Задание
17.	
18.	
19.	
20.	

Варианты с 21 по 35

Вариант	Задание	Вариант	Задание
---------	---------	---------	---------

Вариант	Задание	Вариант	Задание
21.		22.	
23.		24.	
25.		26.	
27.		28.	

Вариант	Задание	Вариант	Задание
29.		30.	
31.		32.	
33.		34.	
35.	