

Рабочая тетрадь

Информатика и ИКТ

Основы программирования на Pascal

2

Ученика _____ класса

Кушнаренко Сергей Николаевич
©КАДЕТСКАЯ ШКОЛА ЛИПЕЦКОЙ ОБЛАСТИ

Оглавление

| | |
|---|----|
| Знакомство с языком Паскаль..... | 3 |
| Практическое задание..... | 5 |
| Ветвления на Паскале. Программирование диалога с компьютером..... | 10 |
| Практическое задание..... | 12 |
| Программирование циклов..... | 18 |
| Практическое задание. Использование циклов в вычислительных алгоритмах..... | 19 |
| Практическое задание. Программирование циклов и ветвлений на Паскале. Алгоритм Евклида..... | 24 |
| Массивы в Паскале..... | 28 |
| Практическое задание. Обработка массивов..... | 32 |

Информатика и ИКТ. Основы программирования на Pascal. Рабочая тетрадь. В 2-х частях. Часть 2. / Автор-составитель: С.Н. Кушнаренок. – Электронное издание (<http://snk.ucoz.ru>) – Липецк, Кадетская школа Липецкой области, 2017 – 36 с.

Знакомство с языком Паскаль

1. Напишите на языке **Паскаль** программу вычисления y по формуле:
 $y = (1 - x^2 + 5x^4)^2$, где x - данное **целое** число.

2. **Перепишите** программу на языке Паскаль, исправив **ошибки**:

```
Program primer  
Var a,b: integer;  
Begin ;  
  ReadLn( a,b);  
  s=a+b;  
  m:=a*b;  
  d:=a:b;  
  WriteLn(s);  
  WriteLn(m);  
  WriteLn(d)  
End
```

3. Не используя функцию возведения в квадрат, напишите на языке Паскаль наиболее **короткие** алгоритмы вычисления:

а) $y=x^{10}$ б) $y=x^{15}$

Постараться использовать **минимальное** количество **дополнительных** переменных.

а)

б)

4. Запишите на языке **Паскаль** программу **циклического обмена** значениями **трёх переменных a,b,c**. Схема циклического обмена: $a \leftarrow b \leftarrow c$

Например, если до обмена было: $a=1, b=2, c=3$, то после обмена должно стать: $a=2, b=3, c=1$.

Практическое задание

1 уровень сложности

1. а) Набрать в редакторе системы Турбо-Паскаль следующую программу:

```
Program my;  
Var a,b,c,s:integer;  
Begin  
  ReadLn(a);  
  ReadLn(b);  
  ReadLn(c);  
  S:=a*b*c;  
  WriteLn(S)  
End.
```

б) Откомпилировать набранную программу и исправить ошибки.

в) Запустить данную программу на выполнение и проверить правильность её работы для чисел 2, 4 и 6.

г) Запустить данную программу на выполнение и проверить правильность её работы для чисел 1, 0 и -1.

д) Запустить данную программу на выполнение и проверить правильность её работы для чисел -2, 3 и 10.

2. Написать программу, которая присваивает целой переменной А значение 10 и выводит это значение на экран.

3. Написать программу, которая запрашивает ввод целого числа в переменную В и выводит это число на экран. Проверить правильность работы программы на числах 1, -5, 256, 10455.

4. Написать программу, которая запрашивает ввод вещественного числа в переменную С, умножает это число на 2 и выводит результат на экран. Проверить правильность работы программы на числах 2.5, -7.33, 0, 782.234.

5. Написать программу для ввода значения величины X целого типа, присваивания величине Y действительного типа значения 5.5, вычисления значения величины $Z = X - Y$ и вывода значения величины Z. Протестировать программу для $X=5.5$, $X=0$, $X=-10.2$

6. Написать программу для ввода значения величины X целого типа, присваивания величине Y действительного типа значения 2.5, вычисления значения величины $Z=X/Y$ и вывода значения величины Z. Протестировать программу для $X=5$, $X=0$, $X=-8.75$

2 уровень сложности

1. Написать на языке Паскаль программу ввода **четырёх целых чисел** и вычисления их **среднего арифметического**. **Протестировать** программу на различных исходных данных (включая вещественные числа) и **доказать** правильность её работы.
2. Вводятся величины **X, Y** **целого** типа. Написать программу для **обмена** значений величин. Необходимо использовать **вспомогательную** величину **T**. **Протестировать** программу для **X=5** и **Y=-11**.
3. Написать программу для вычисления **дискриминанта d** квадратного уравнения **$ax^2 + bx + c = 0$** . **Разработать** тесты проверки правильности работы программы для вариантов, когда **d>0**, **d=0** и **d<0**.
4. Из железной полосы длиной **L** метров нужно изготовить обруч. На соединение концов уходит **D** метров полосы. Написать программу для вычисления **радиуса R** обруча. **Протестировать** программу для а) **L=5.8, D=0.2**, б) **L=3.25, D=0.1**
5. Найти **площадь кольца**, внешний радиус которого равен **R₁**, а внутренний – **R₂** ($R_1 > R_2$). **Протестировать** программу для **R₁=5.6** и **R₂=3.8**. **Проверить** ответ на калькуляторе.
6. Написать на языке Паскаль программу для **вычисления** выражения:

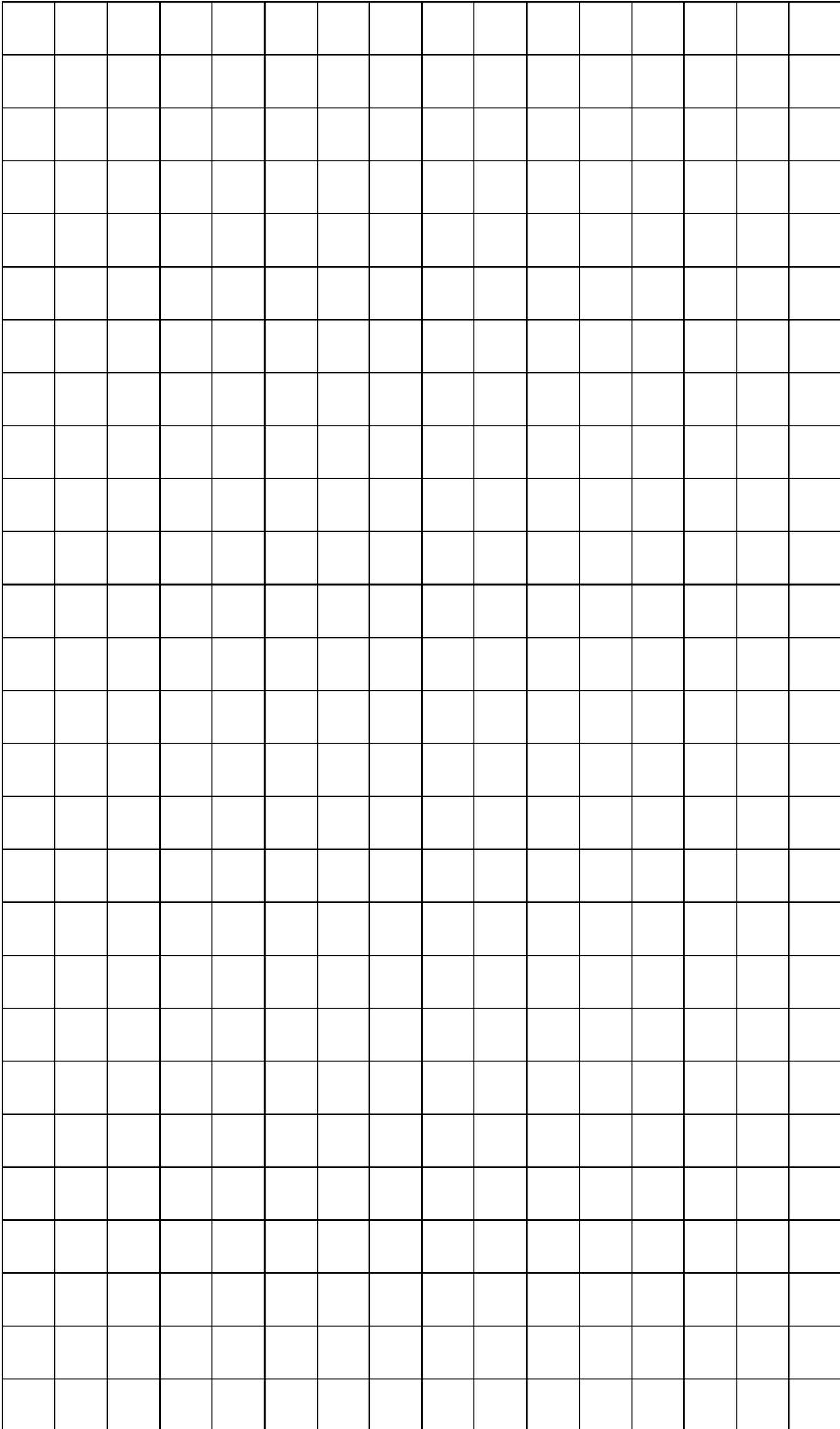
$$S = (2x+y)(x-y)$$

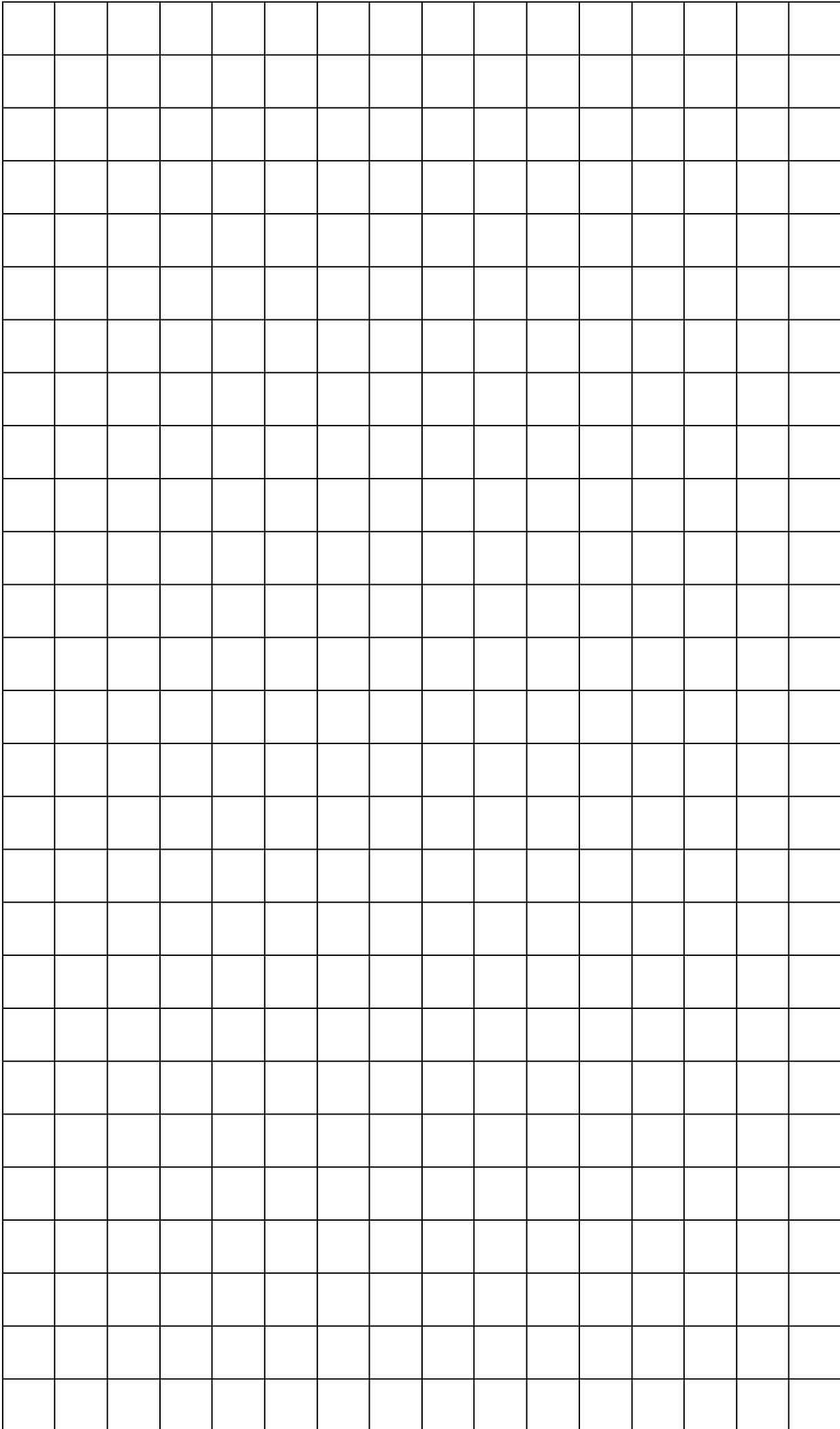
Протестировать её для следующих исходных данных:

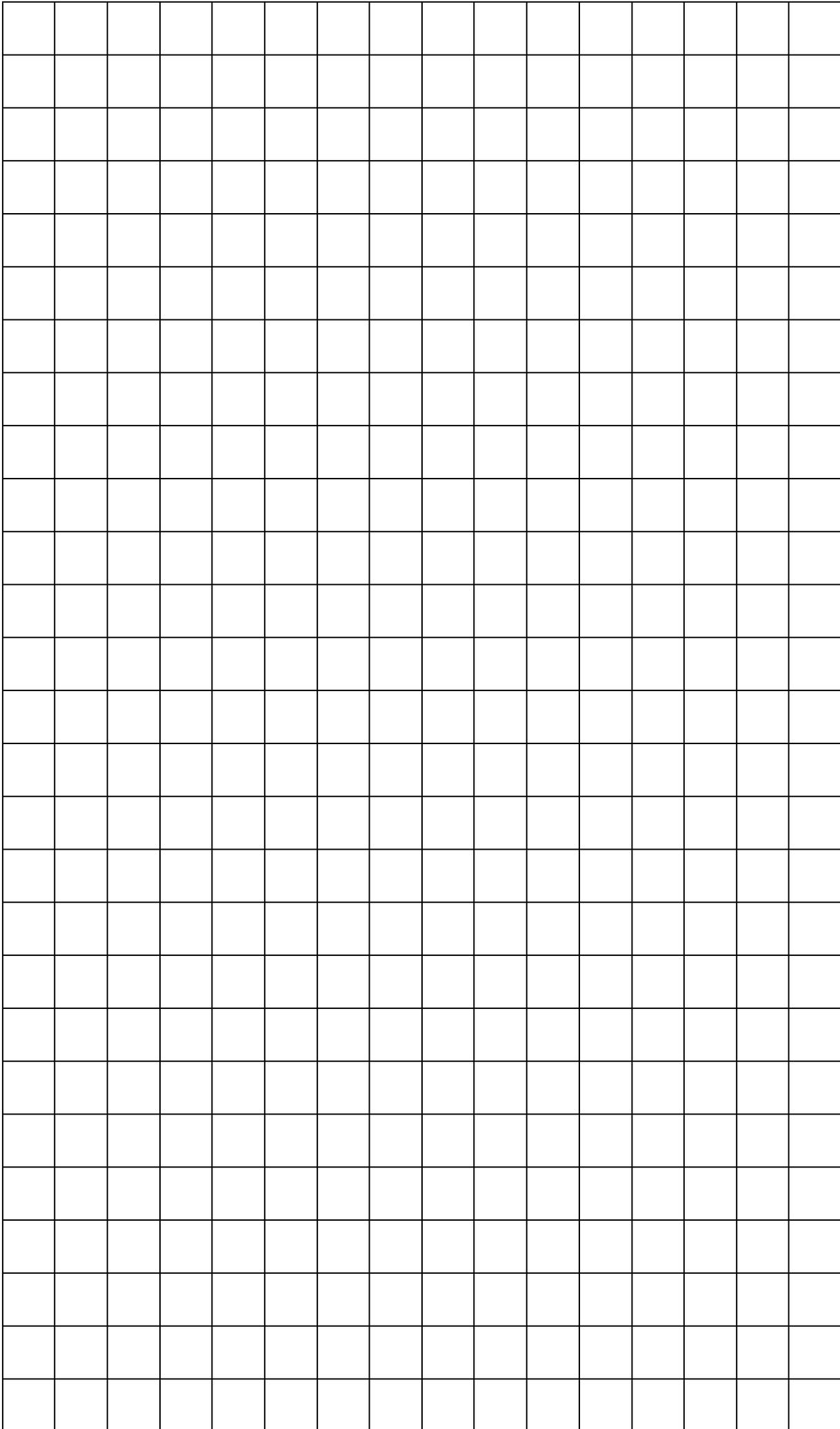
1) **x=2, y=1** 2) **x=3, y=0** 3) **x=0, y=-2**

3 уровень сложности

1. Заданы величины **X, Y** **действительного** типа. Написать программу для **обмена** значений величин. Использовать **вспомогательные** величины **нельзя**. **Протестировать** программу для **X=-3** и **Y=8**.
2. Дано **натуральное** число **X**. Вычислить **$Y = X^5$** . Разрешается использовать только **три** операции **умножения**. **Протестировать** программу для **X=-2** и **X=3**.
3. Дано **натуральное** число **X**. Вычислить **$Y = 1 - 2X + 3X^2 - 4X^3$** . Разрешается использовать **не более 8** арифметических операций. Допустимы: операции сложение, вычитание, умножение. **Протестировать** программу для **X=0, X=1, X=-2**.
4. Вычислить **расстояние** между двумя точками с координатами **(X₁, Y₁)** и **(X₂, Y₂)**. **Доказать** правильность работы программы на **трёх** различных тестах.







Ветвления на Паскале. Программирование диалога с компьютером

1. Составьте на **Паскале** программу определения **минимума** из **трёх** чисел с использованием **логических операций**. Программа должна иметь **дружественный интерфейс**.

2. Составьте на **Паскале** программу, которая запрашивает **возраст двух детей** (Саши и Маши), а затем выводит информацию о том, кто из них **старше**.

3. Запишите на языке **Паскаль** программу, которая по **номеру для недели** будет выводить его **название**.

4. Что **вычисляет** приведённая ниже программа?

```
Program example;  
Var x,y,z,l :integer;  
Begin  
  ReadLn(a,b,c);  
  i:=0;  
  If a>0 then i:=i+1;  
  If b>0 then i:=i+1;  
  If c>0 then i:=i+1;  
  WriteLn(i)  
End.
```

Практическое задание

1 уровень сложности

1. Дано **целое** число **A**. Если значение **A > 0**, то необходимо **увеличить** его на **единицу**. Разработать схему алгоритма для решения этой задачи. **Протестировать** алгоритм для **A=5, A=-4, A=0**.

2. Дано **целое** число **A**. Если значение **A < 0**, то необходимо **удвоить** его. Разработать схему алгоритма для решения этой задачи. **Протестировать** алгоритм для **A=6, A=-10, A=0**.

3. Дано **целое** число **A**. Если значение **A <> 0**, то необходимо **уменьшить** его на **4**. Написать программу для решения этой задачи. **Протестировать** алгоритм для **A=2, A=-1, A=0**.

4. Дано **целое** число **A**. Если значение **A > 0**, то необходимо **увеличить** его на **единицу**, иначе **уменьшить** на **1**. Разработать схему алгоритма для решения этой задачи. **Протестировать** алгоритм для **A=3, A=0, A=-12**.

5. Дано **целое** число **A**. Если значение **A = 0**, то необходимо **увеличить** его на **3**, иначе присвоить **A** значение, **равное 0**. Разработать схему алгоритма для решения этой задачи. **Протестировать** алгоритм для **A=0, A=-1, A=8**.

6. Даны два **действительных** числа **X** и **Y**. Если **X > Y**, то вычислить **произведение** этих чисел, иначе их **сумму**. Разработать схему алгоритма для решения этой задачи. **Протестировать** алгоритм на **трёх** различных тестах (**X > Y**, **X = Y** и **X < Y**).

2 уровень сложности

1. Даны два **действительных** числа **X** и **Y**, не равные друг другу. **Заменить меньшее** из этих чисел половиной их суммы, а **большее** – их удвоенным произведением. Разработать схему алгоритма для решения этой задачи. **Протестировать** алгоритм на числах **5.5** и **-4.3**, а также на числах **1** и **14.5**.

2. Точка **A** задана координатами **X, Y**. Разработать схему алгоритма, который устанавливает значение флага **F=1**, если точка принадлежит заштрихованной области (см. рисунок 1) и значение флага **F=0** в противном случае. Вывести значение **F**. **Протестировать** алгоритм для точек **(1.5, 2)**, **(0, 0)**, **(-1.5, 1)**, **(1, -1.2)**, **(-2, -1)**.

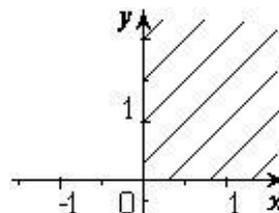


Рис.1

3. Точка **A** задана координатами **X, Y**. Разработать схему алгоритма, который устанавливает значение флага **F=1**, если точка принадлежит заштрихованной области (см. рисунок 2) и значение флага **F=0** в противном случае. Вывести значение **F**. **Протестировать** алгоритм для точек **(2.5, 2)**, **(1, 1)**, **(0, 0)**, **(1, 0)**, **(2, -1)**.

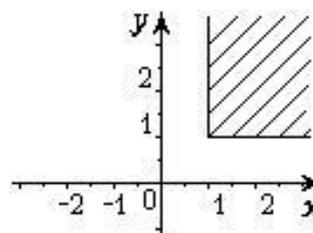


Рис.2

4. Точка **A** задана координатами **X,Y**. Разработать схему алгоритма, который устанавливает значение флага **F=1**, если точка принадлежит заштрихованной области (см. рисунок 3) и значение флага **F=0** в противном случае. Вывести значение **F**. **Протестировать** алгоритм для точек **(0,0.8)**, **(0,0)**, **(-1.5,1)**, **(1,1.5)**, **(-2,-1)**.

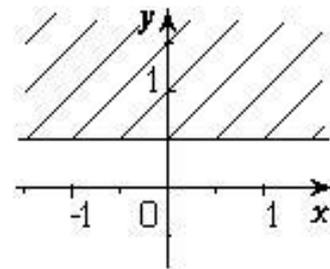


Рис.3

5. Разработать схему алгоритма для определения **минимума** из **трёх** чисел **без использования логических операций**. **Протестировать** алгоритм на следующих исходных данных:

- а) **2 5 1**
- б) **0 -2 8**
- в) **-4 5 10**

6. Разработать схему алгоритма для подсчета количества **отрицательных** чисел среди **целых** чисел **a, b, c**. **Протестировать** алгоритм для **всех** возможных случаев (когда количество отрицательных чисел равно 0, 1, 2 и 3).

3 уровень сложности

1. **Треугольник** задан длинами сторон **A, B, C**. Разработать схему алгоритма, определяющую, **существует** ли данный треугольник. Если треугольник существует, то установить значение флага **F=1**, иначе **F=0**. Для решения этой задачи использовать **сложные логические условия**. **Протестировать** алгоритм для следующих исходных данных:

- а) **A=3, B=4, C=5**
- б) **A=1, B=1, C=1**
- в) **A=0, B=4, C=5**
- г) **A=-3, B=6, C=5**
- д) **A=2, B=1, C=8**

2. Разработать схему алгоритма для отыскания **max(min(a,b), min(c,d))**, **не используя** сложные логические условия и вложенные ветвления. Числа **a,b,c,d** - целые. **Протестировать** алгоритм для следующих исходных данных:

- а) **a=4 b=5 c=6 d=9**
- б) **a=2 b=1 c=6 d=9**
- в) **a=2 b=1 c=8 d=4**
- г) **a=12 b=1 c=6 d=9**

3. Точка **A** задана координатами **X,Y**. Разработать схему алгоритма, который устанавливает значение флага **F=1**, если точка принадлежит заштрихованной области (см. рисунок 4) и значение флага **F=0** в противном случае. Вывести значение **F**. **Протестировать** алгоритм для точек **(0,0)**, **(1,0)**, **(1.5,1)**, **(-1,1.5)**, **(-2,-1)**, **(2,-1)**, **(1,-1)**, **(-1,1)**.

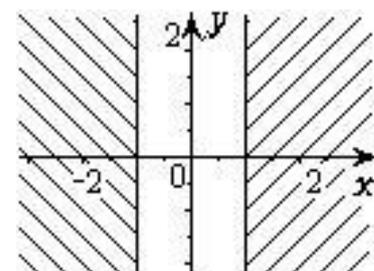
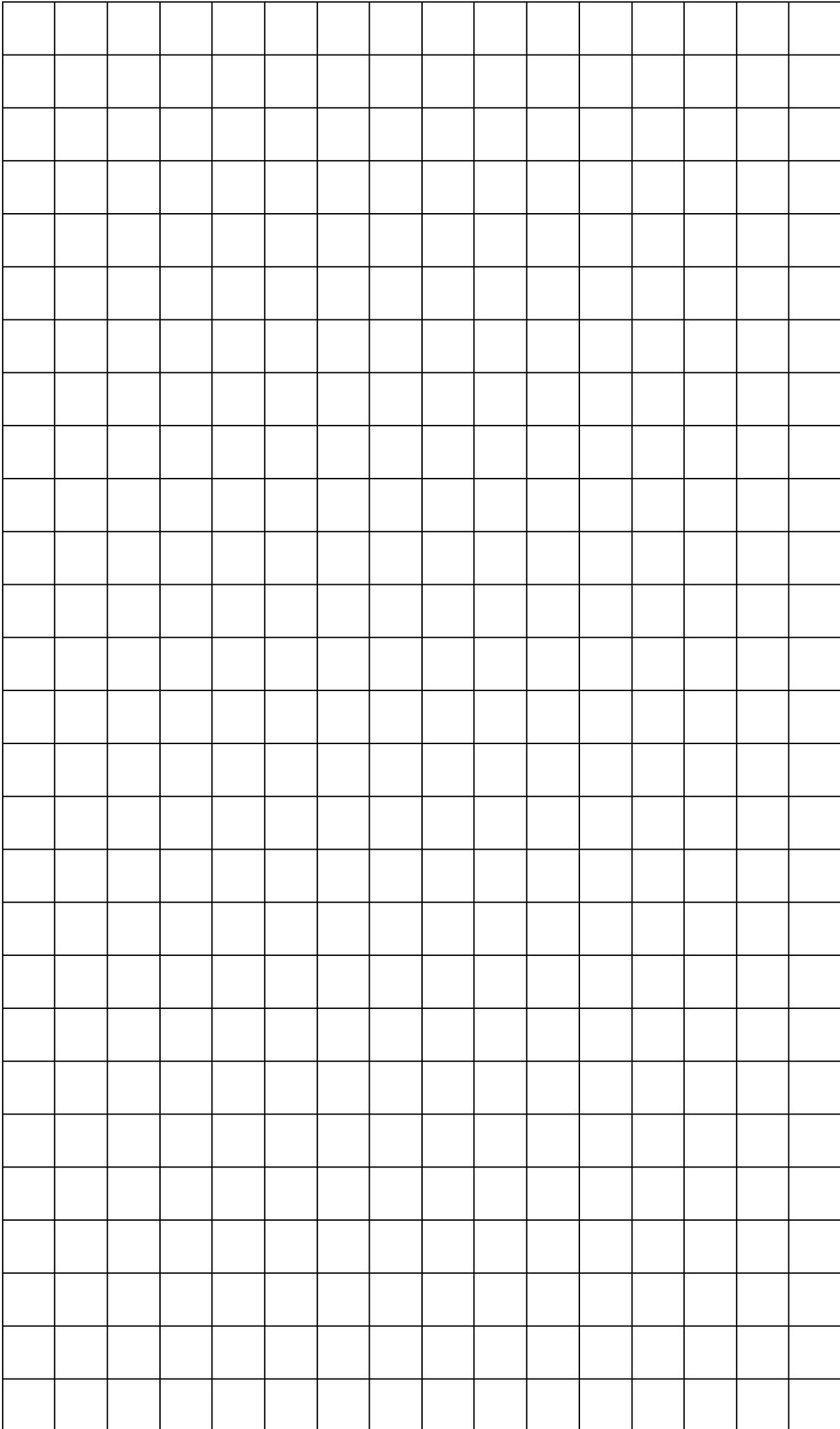
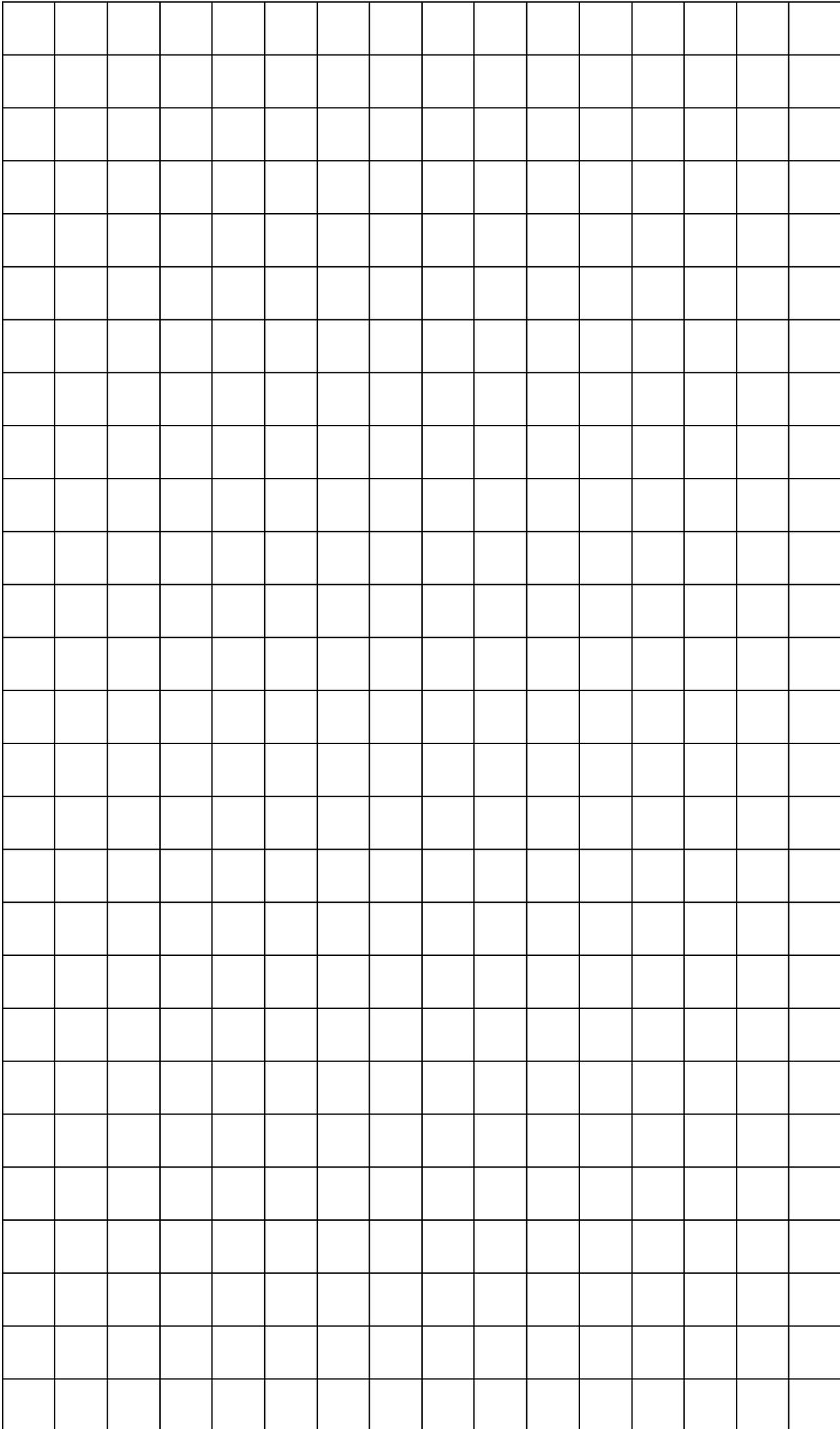
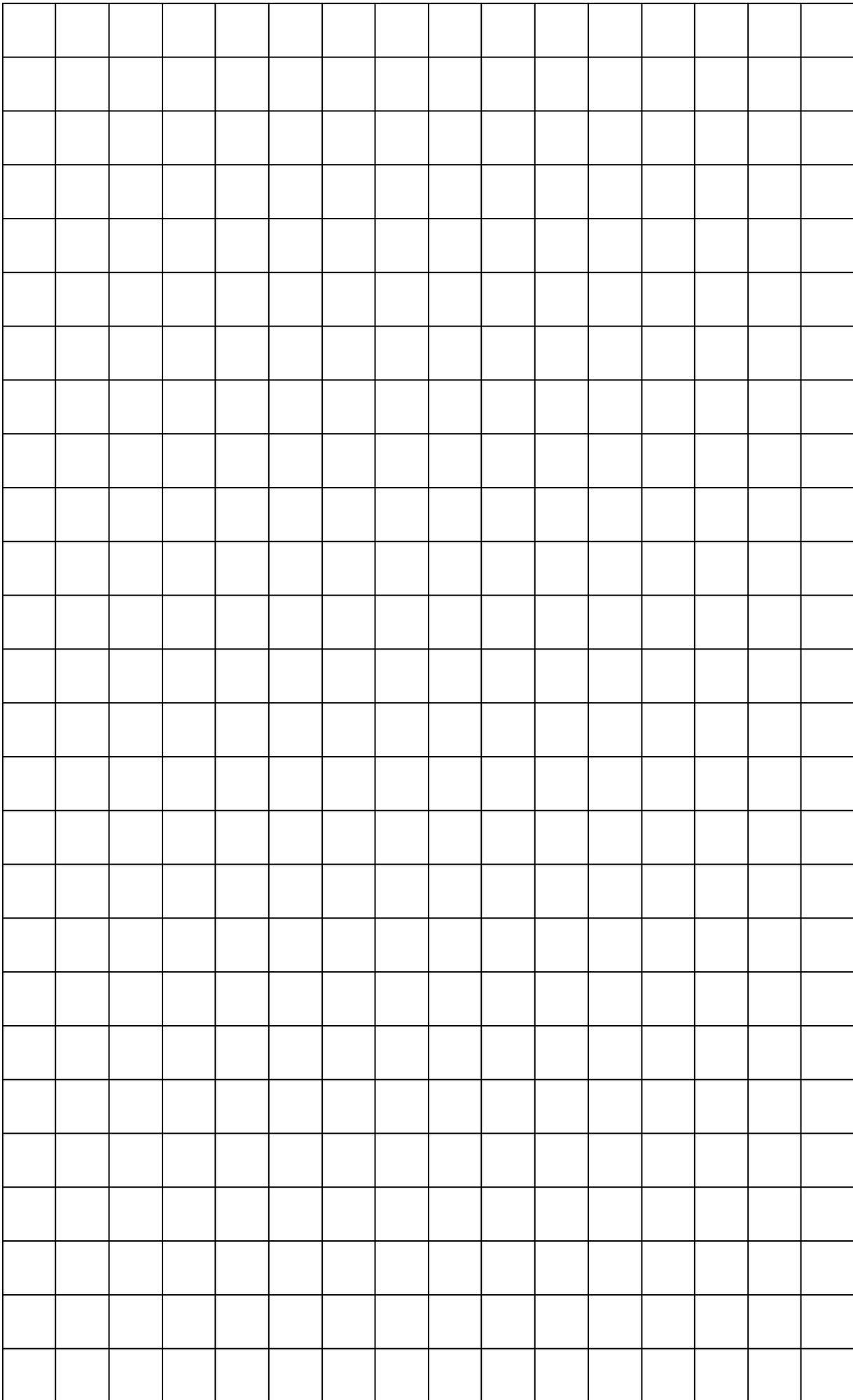


Рис. 4



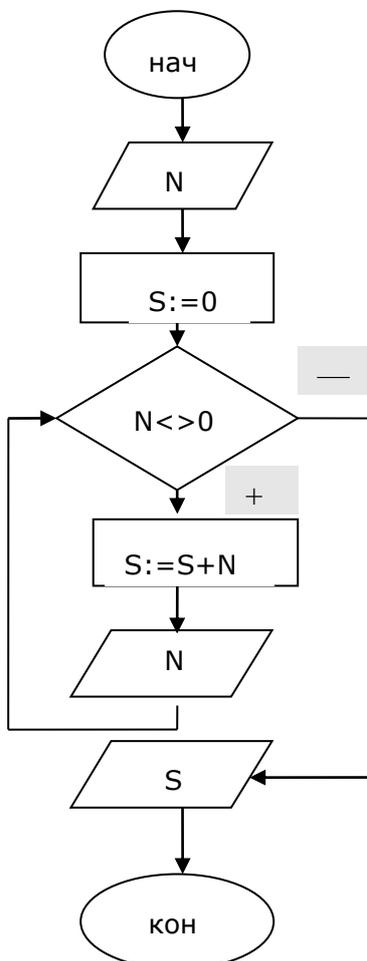




Программирование циклов

1. Составьте на **Паскале** программу возведения в **N-ую** степень **целого** числа **X**.

2. Запишите на языке **Паскаль** программу, соответствующую приведённой ниже **блок-схеме** и определить, что вычисляет данная программа:



3. Составьте на языке **Паскаль** программу вычисления **суммы** всех **натуральных** чисел, **не превышающих** заданного натурального числа **N**. Построить **трассировочную** таблицу.

Практическое задание. Использование циклов в вычислительных алгоритмах

1 уровень сложности

1. Разработать схему алгоритма, который запрашивает **N произвольных** целых чисел и ищет их **сумму**. Число **N** **вводится** с клавиатуры. **Разработать** для созданного алгоритма не менее **трёх** тестов и проверить на них правильность работы алгоритма.

2. Разработать схему алгоритма для вычисления **суммы ряда**:

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{N}$$

Число **N** **вводится** с клавиатуры.

Протестировать алгоритм для **N=1, N=2, N=4, N=6**. Правильность ответов проверить с помощью калькулятора.

3. Одноклеточная амёба каждые **три часа** делится на **2** клетки. Разработать схему алгоритма для определения, сколько амёб будет через **3, 6, 9, 12** и **24** часа.

4. Дано **натуральное** число **N** и **действительное** число **x**. Разработать схему алгоритма для вычисления **суммы** ряда:

$$\sin x + \sin 2x + \sin 3x + \dots + \sin N \cdot x$$

Протестировать алгоритм для **x=1.5** и **N=3**. Проверить правильность ответа

на калькуляторе.

5. Разработать схему алгоритма для вычисления N -ой степени числа a (a^N). Протестировать алгоритм для а) $a=3$ и $N=4$, б) $a=2$ и $N=5$.

2 уровень сложности

1. Начав тренировки, спортсмен в **первый** день пробежал **10** км. Каждый день он **увеличивал** дневную норму на **10%** нормы предыдущего дня. Какой **суммарный** путь пробежит спортсмен за N дней? Разработать схему алгоритма для решения этой задачи. Протестировать алгоритм для $N=3$ и $N=7$.

2. Ежемесячная стипендия студента составляет A рублей в месяц, а расходы на проживание превышают стипендию и составляют B рублей в месяц. Рост цен ежемесячно **увеличивает** расходы на **3%**. Разработать схему алгоритма для расчёта необходимой суммы денег, которую надо единовременно попросить у родителей, чтобы можно было прожить учебный год (**10** месяцев), используя только эти деньги и стипендию. Протестировать алгоритм для следующих исходных данных:

а) $A=1000, B=1100$

б) $A=900, B=1000$

в) $A=600, B=1200$

3. Разработать схему алгоритма для нахождения всех **делителей натурального** числа N . Протестировать алгоритм для $N=10, N=75, N=99, N=13$.

4. Разработать схему алгоритма для вычисления $N!$ (факториал числа N). Факториал вычисляется по формуле:

$$N! = \begin{cases} 1 * 2 * 3 * 4 * \dots * N, N > 0 \\ 1, N = 0 \end{cases}$$

Операцию вычисления факториала использовать нельзя!

Протестировать алгоритм для $N=0, N=2, N=4$.

3 уровень сложности

1. Дано **натуральное** число N . Разработать схему алгоритма для вычисления суммы ряда: $S=1!+2!+3!+\dots+N!$ ($N>1$)

Операцию вычисления факториала использовать нельзя!

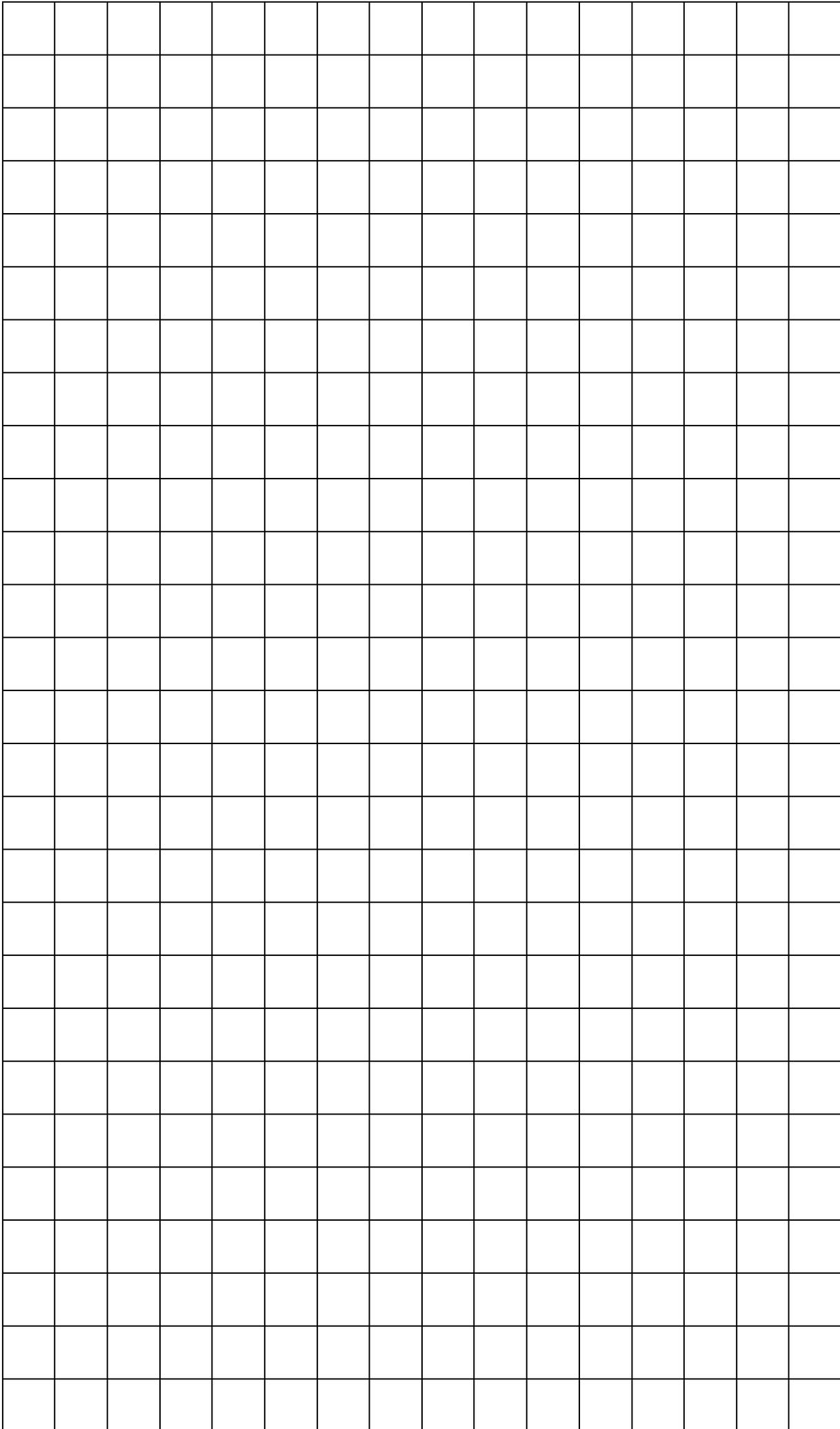
Протестировать алгоритм для $N=3$ и $N=5$

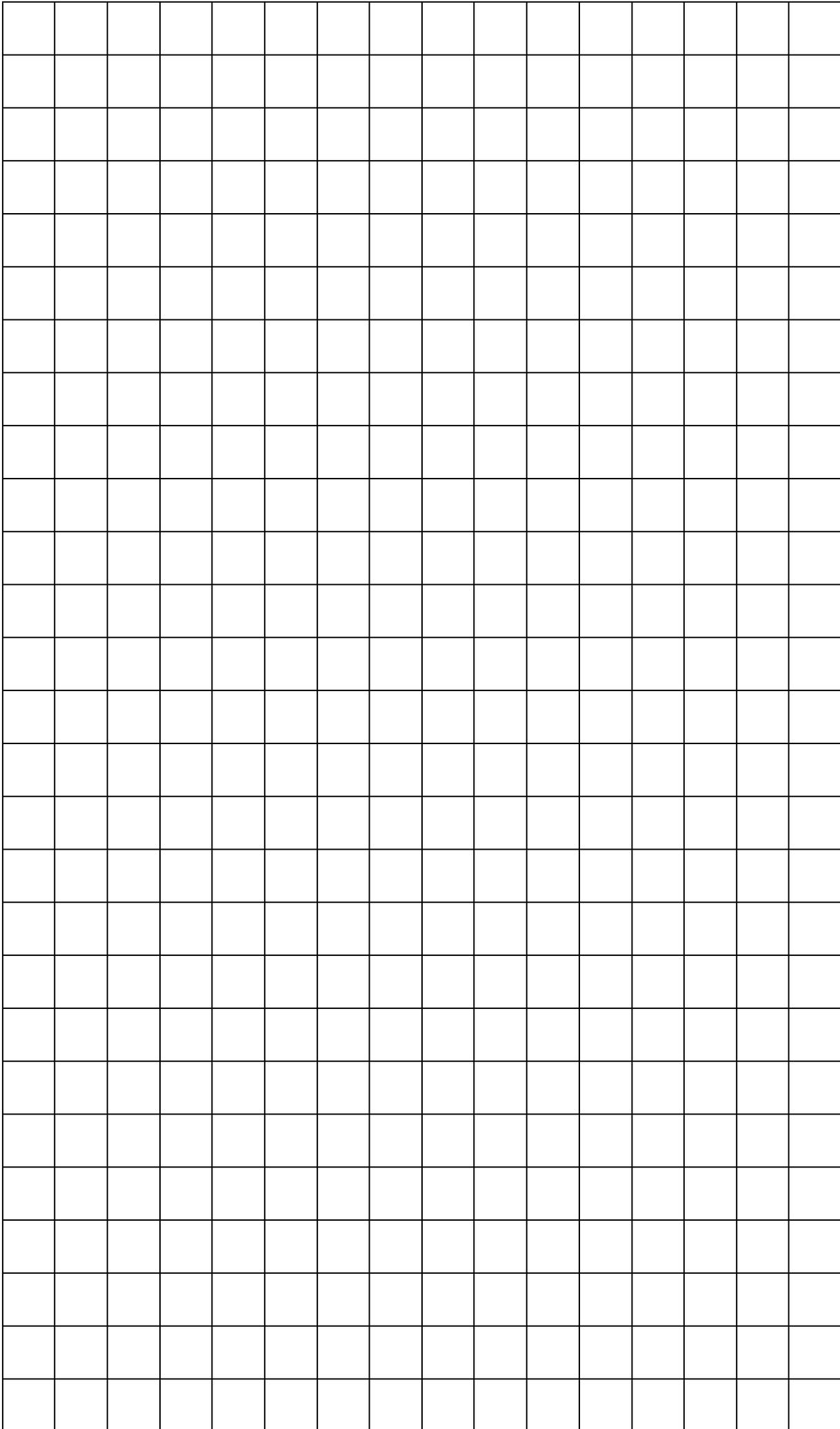
2. Покупатель должен заплатить в кассу S рублей. У него имеются монеты в **1, 2** и **5** рублей, а также купюры достоинством в **10, 50, 100** и **500** рублей (достаточное количество). Сколько монет и купюр разного достоинства отдаст покупатель, если он начинает платить с самых крупных? Операции **целочисленного** деления (div и mod) использовать **нельзя**. Разработать схему алгоритма для решения этой задачи. Протестировать алгоритм для $S=567, S=1025, S=64, S=13$.

3. Числа **Фибоначчи** (f_n) определяются формулами:

$$f_0=f_1=1, f_n=f_{n-1}+f_{n-2} \text{ при } n=2,3,\dots$$

Разработать схему алгоритма для вычисления числа Фибоначчи. Опреде-





Практическое задание. Программирование циклов и ветвлений на Паскале. Алгоритм Евклида

1 уровень сложности

1. Набрать и отладить программу нахождения наибольшего общего делителя и протестировать её на следующих исходных данных:

а) $a=40$ $b=64$

б) $a=32$ $b=16$

в) $a=12$ $b=20$

г) $a=5$ $b=12$

2. Составить на языке Паскаль программу, которая запрашивает с клавиатуры N целых чисел и считает, сколько из них положительных, сколько отрицательных и сколько нулей. Разработать для программы не менее двух тестов и доказать правильность её работы.

3. Написать программу, которая выводит на экран все двухзначные числа, делящиеся на 3 или на 5 (использовать операции целочисленного деления div или mod).

2 уровень сложности

1. Составить программу нахождения наименьшего общего кратного (НОК) двух чисел m и n , используя формулу $n*m=\text{НОД}(n,m)*\text{НОК}(n,m)$. Разработать тесты для проверки правильности работы программы.

2. Проверить, являются ли два числа a и b взаимно простыми. Два числа называются взаимно простыми, если их наибольший общий делитель равен 1. Протестировать программу для следующих исходных данных:

а) $a=5$ $b=95$

б) $a=3$ $b=100$

в) $a=11$ $b=98$

3. Написать программу сокращения дроби $\frac{N}{M}$, где N – целое, M – натуральное число. Использовать алгоритм Евклида. Разработать тесты для проверки правильности работы программы.

3 уровень сложности

1. Составить на языке Паскаль программу нахождения наибольшего общего делителя трёх чисел, используя следующую формулу: $\text{НОД}(a,b,c)=\text{НОД}(\text{НОД}(a,b),c)$. Протестировать программу на следующих исходных данных:

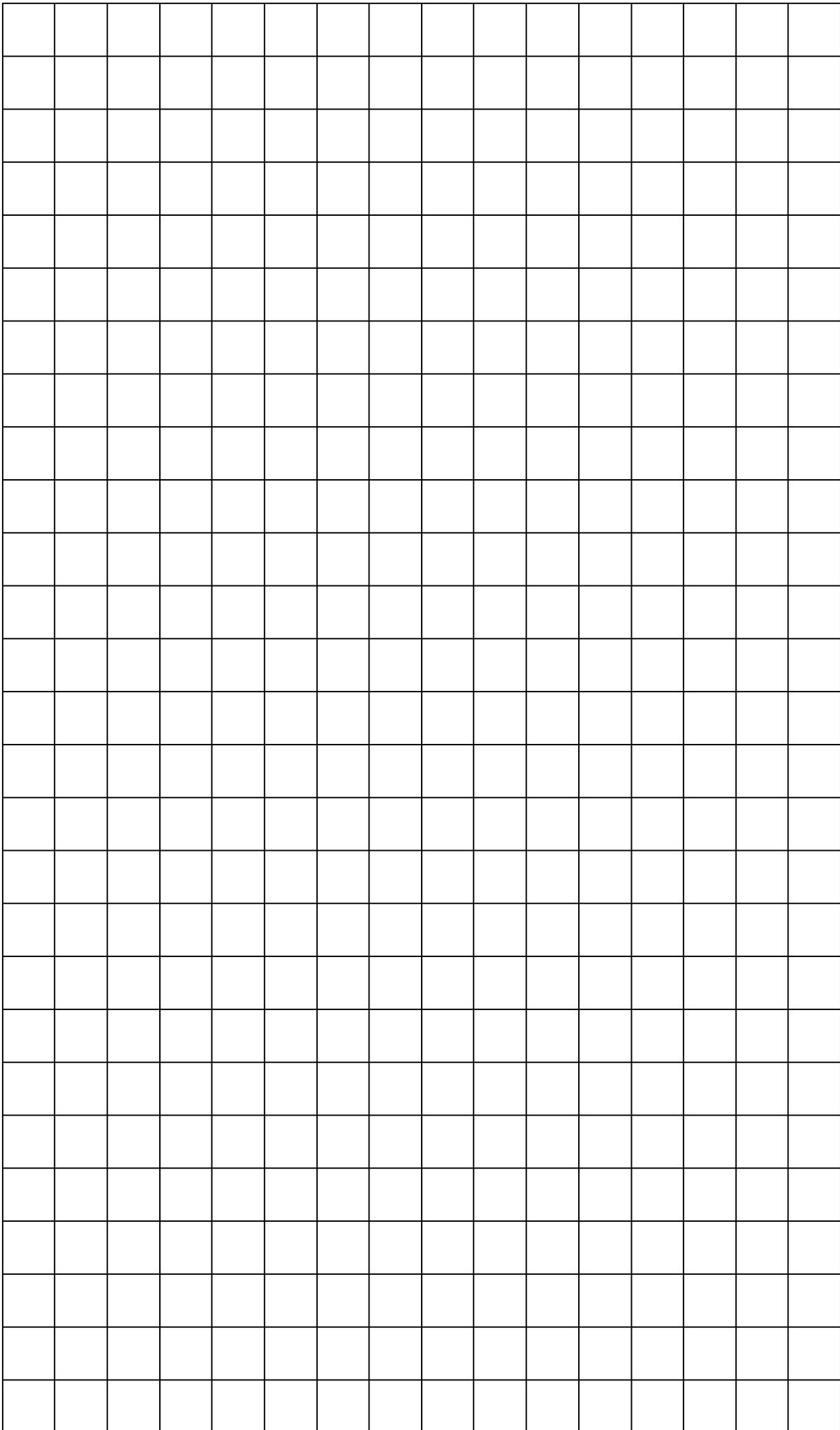
а) $a=40$ $b=64$ $c=120$

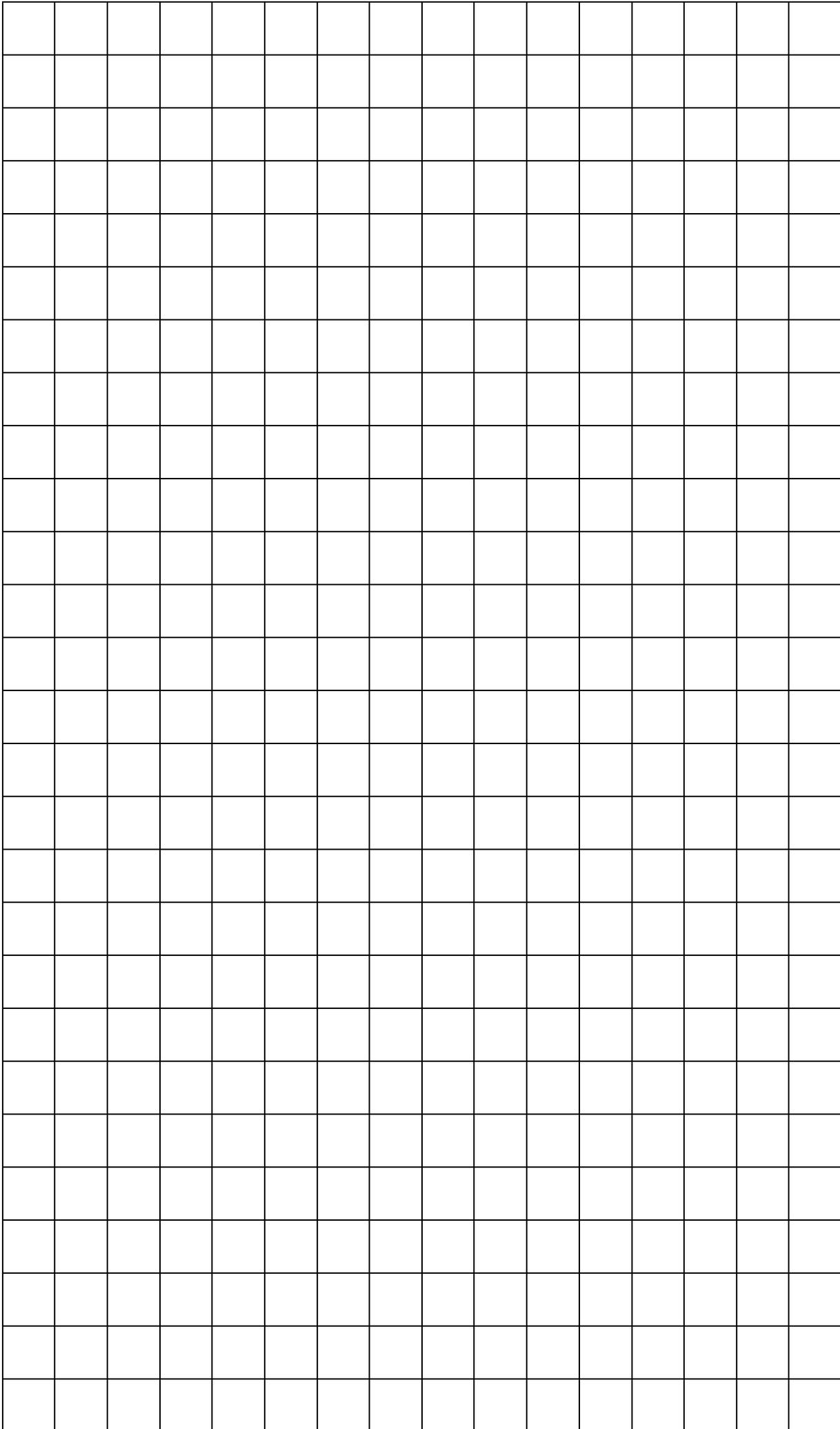
б) $a=32$ $b=45$ $c=60$

в) $a=11$ $b=220$ $c=33$

г) $a=175$ $b=1025$ $c=600$

2. Написать программу, запрашивающую с клавиатуры целые числа и определяющую максимальное и минимальное из них. Окончание ввода –





Массивы в Паскале

1. Вы посетили магазин и купили 10 видов товара. В таблицу T[1:10] вы записали количество купленного товара каждого вида. В таблицу C[1:10] записали цены единиц каждого вида товара соответственно. Составьте на Паскале программу вычисления общей стоимости всех покупок.

2. Вы приобрели котёнка. Каждый вечер вы определяете его вес с помощью весов. Как можно описать на Паскале массив, в котором будут храниться значения веса котёнка в течение одного месяца (например, мая)?

3. Напишите на Паскале программу, которая запрашивает с клавиатуры массив из N чисел, а затем выводит среднее арифметическое всех элементов массива.

4. Перепишите программу на Паскале, исправив ошибки:

```
Program massiv;  
Var a:array(-10..10) of integer;  
Begin  
  For i:=1 to 20 Do  
    Read(a[i]);  
  For i:=1 to 20 Do  
    a[i]:=a[i]/10;  
  For i:=1 to 20 Do  
    Write(a[i])  
End.
```

Сформулируйте условие этой задачи

Практическое задание. Обработка массивов

1 уровень сложности

1. Разработать схему алгоритма, который вводит массив из N целых чисел и выводит на экран этот же массив в прямом и обратном порядке. Протестировать алгоритм на произвольных массивах, состоящих из 1 числа, из 5 чисел, из 10 чисел.

2. Разработать схему алгоритма, который вводит массив из N целых чисел и выводит номера отрицательных элементов и сами эти элементы. Протестировать алгоритм для следующих массивов:

а) 3 5 -2 3 -2 0 -6 -8 1

б) -1 -2 -3 -4 0 -1 2 3

3. Разработать схему алгоритма, который вводит массив из N целых чисел и выводит на экран элементы с чётными номерами. Протестировать алгоритм на произвольных массивах размерностью 5 и 8 элементов.

4. Разработать схему алгоритма, который вводит массив из N целых чисел и выводит на экран сам массив и сумму всех его элементов. Протестировать алгоритм на следующих массивах:

а) 1 3 4 -2

б) 0 1 -2 10 11 12 -10 -3

в) 1 1 1 1 -1 -1 -1 -1

2 уровень сложности

1. Разработать схему алгоритма, который запрашивает массив из N целых чисел, а затем складывает все элементы с нечётными номерами и все элементы с чётными номерами и выводит их суммы, а также сам исходный массив.

Пример:

Массив: 6 3 5 1 1 3

Суммы: $6+5+1=12$ (элементы с нечётными номерами) и $3+1+3=7$ (элементы с чётными номерами).

Протестировать алгоритм на приведённом выше примере, а также на двух других произвольных тестах.

2. Разработать схему алгоритма, который вводит массив из N целых чисел и выводит на экран элементы с нечётными номерами в обратном порядке. Протестировать алгоритм для следующих исходных данных:

а) 1 3 5 6 8 9

б) 1 4 6 2 4 6 8 6 9

3. У прилавка магазина выстроилась очередь из N покупателей. Время обслуживания i -го покупателя равно t_i . Определить время S_i пребывания i -го покупателя в очереди. Разработать схему алгоритма для решения этой задачи. Протестировать алгоритм на следующих исходных данных:

а) Номер покупателя – 5. Время обслуживания отдельных покупателей (в минутах): 1 1.5 2 1.5 2.5 3 1.5 1

