Казанский (Приволжский) федеральный университет

Сибайский институт Уфимского университета науки и технологий

**Практикум по основам программирования**

**учебное пособие для бакалавров и магистрантов ВУЗов,   
специализирующихся в области информационных**

**технологий и программирования**

**Казань - Сибай - 2023**

Арабов М.К., Гумеров И. С.

Практикум по основам программирования: учебное пособие для бакалавров и магистрантов ВУЗов, специализирующихся в области информационных технологий и программирования /М.К. Арабов, И.С. Гумеров. - \_\_\_\_\_\_\_\_\_\_\_\_, 2023. 342 с.

*В учебном пособии расмотрены основные методы и приемы программирования на самых востребованных и современных языках программирования, таких как PascalABC.NET, C++, Python, C#, Java, Go, Kotlin и Visual Basic.NET. Данное пособие можно использовать для приобретения практических навыков в области алгоритмизации и программирования на современных языках программирования. В нем приводятся большое число типовых алгоритмов для решения задач различных классов. Пособие можно использовать при проведении практических занятий по дисциплинам, связанным с программированием, таким как «Практикум по программированию», «Основы алгоритмизации и языки программирования», «Теория алгоритмов», «Методы программирования», «Основы информатики и программирования», «Программная инженерия» и т.п.*

**Редактор:**

**Рецензенты**

© Арабов М.К., Гумеров И. С – 2023.

Содержание

[Предисловие 5](#_1fob9te)

[1. Ввод и вывод данных, оператор присваивания: группа Begin 7](#_3znysh7)

[2. Целые числа: группа Integer 18](#_2et92p0)

[Задачи для самостоятельной работы 34](#_3dy6vkm)

[3. Условный оператор: группа If 46](#_1t3h5sf)

[4. Оператор выбора: группа Case 62](#_4d34og8)

[Задачи для самостоятельной работы 77](#_2s8eyo1)

[5. Цикл с параметром: группа For 97](#_17dp8vu)

[Вложенные циклы 110](#_3rdcrjn)

[6. Цикл с условием: группа While 113](#_26in1rg)

[7. Последовательности: группа Series 129](#_lnxbz9)

[Вложенные циклы 134](#_35nkun2)

[Задачи для самостоятельной работы 138](#_1ksv4uv)

[8. Процедуры и функции: группа Proc 151](#_44sinio)

[1. Процедуры с числовыми параметрами 151](#_2jxsxqh)

[2. Функции с числовыми параметрами 156](#_z337ya)

[3. Дополнительные задания на процедуры и функции 160](#_3j2qqm3)

[Задачи для самостоятельной работы 163](#_1y810tw)

[9. Одномерные массивы: группа Array 169](#_4i7ojhp)

[1. Формирование массива и вывод его элементов 169](#_2xcytpi)

[2. Анализ элементов массива 175](#_1ci93xb)

[3. Работа с несколькими массивами 181](#_3whwml4)

[4. Преобразование массива 183](#_2bn6wsx)

[a) Изменение элементов массива 183](#_qsh70q)

[b) Удаление и вставка элементов 185](#_3as4poj)

[c) Сортировка массива 188](#_1pxezwc)

[5. Серии целых чисел 190](#_49x2ik5)

[6. Множества точек на плоскости 194](#_2p2csry)

[10. Двумерные массивы (матрицы): группа Matrix 198](#_147n2zr)

[1. Формирование матрицы и вывод ее элементов 198](#_3o7alnk)

[2. Анализ элементов матрицы 204](#_23ckvvd)

[3. Преобразование матрицы 209](#_ihv636)

[4. Диагонали квадратной матрицы 217](#_32hioqz)

[Задачи для самостоятельной работы 223](#_1hmsyys)

[11. Символы и строки: группа String 235](#_41mghml)

[1. Символы и их коды. Формирование строк 235](#_2grqrue)

[2. Посимвольный анализ и преобразование строк. Строки и числа 239](#_vx1227)

[3. Обработка строк с помощью стандартных функций. Поиск и замена 243](#_3fwokq0)

[4. Анализ и преобразование слов в строке 247](#_1v1yuxt)

[5. Дополнительные задания на обработку строк 253](#_4f1mdlm)

[Задачи для самостоятельной работы 259](#_2u6wntf)

[12. Двоичные (типизированные) файлы: группа File 264](#_19c6y18)

[1. Основные операции с двоичными файлами 264](#_3tbugp1)

[a) Создание файла, ввод и вывод его элементов 264](#_28h4qwu)

[b) Преобразование файла 276](#_nmf14n)

[c) Работа с нетипизированными двоичными файлами 280](#_37m2jsg)

[2. Обработка нескольких числовых файлов. Файлы-архивы 281](#_1mrcu09)

[3. Cимвольные и строковые файлы 284](#_46r0co2)

[4. Использование файлов для работы с матрицами 288](#_2lwamvv)

[13. Текстовые файлы: группа Text 291](#_111kx3o)

[1. Основные операции с текстовыми файлами 291](#_3l18frh)

[2. Анализ и форматирование текста 298](#_206ipza)

[3. Текстовые файлы с числовой информацией 303](#_4k668n3)

[4. Дополнительные задания на обработку текстовых файлов 308](#_2zbgiuw)

[Задачи для самостоятельной работы 310](#_1egqt2p)

[14. Составные типы данных в процедурах и функциях: группа Param 317](#_3ygebqi)

[1. Одномерные и двумерные массивы 317](#_2dlolyb)

[2. Строки 324](#_sqyw64)

[3. Файлы 327](#_3cqmetx)

[4. Записи 330](#_1rvwp1q)

[Задачи для самостоятельной работы 333](#_4bvk7pj)

[Литература 341](#_2r0uhxc)

## **Предисловие**

*Я всегда брался за новый язык, чтобы*

*овладеть новым рабочим орудием.*

*Мирче Элиаде[[1]](#footnote-0)*

Пособие является частью методического комплекса, предназначенного для обучения основам программирования на самых востребованных языках программирования. По каждой представленной в пособии теме приводятся примеры решения задач на разных языках программирования и предлагаются задания для самостоятельного решения.

Многие годы в разных вузах в качестве языка обучения программированию используются PascalABC.NET и C++, которые являются ясными, логичными и гибкими языками и приучают к хорошему стилю программирования. Кроме того, в средней школе основы программирования преимущественно преподают именно на базе PascalABC.NET и C++. Но в последние годы стали популярными и другие языкы программирования, такие как Java, C#, VB.NET, Python, Go и Kotlin, и в разных вузах их начинают внедрять в образовательный процесс. Большинство из вышепричисленных языков программирования имеют простые и мощные интегрированные среды разработки, поддерживающие технологию IntelliSense.

В настоящее время существует множество подходов к изучению программирования. Но невозможно изучать программирование на каком-либо языке, не изучив предварительно методы разработки алгоритмов. Поэтому автор попытался написать пособие по алгоритмизации и программированию. Насколько ему это удалось, судить читателю.

В пособии приведены сотни задач с подробным решением на разных языках программирования, что имеет важное практическое значение. Эти задачи можно также использовать для подготовки студентов и школьников для участия в олимпиадах.

Пособие можно использовать для приобретения практических навыков в области алгоритмизации и программирования на современных языках программирования. В нем приводятся основные приемы и методы программирования, большое число типовых алгоритмов для решения задач различных классов, а также примеры их использования. Пособие можно использовать при проведении практических занятий по дисциплинам, связанным с программированием, таким как «Практикум по программированию», «Основы алгоритмизации и языки программирования», «Теория алгоритмов», «Методы программирования», «Основы информатики и программирования», «Программная инженерия» и т.п.

Отметим, что при работе над этим пособием была использована следующая литература [1-27], основными источниками задач послужили [1-6].

Авторы надеются, что читатель имеет первоначальные навыки работы на персональном компьютере под управлением ОС Windows или Linux и знаком со школьным курсом математики.

Мы надеемся, что это пособие поможет читателям не только понять фундаментальные аспекты программирования, но и оценить красоту и изящество этой области человеческой деятельности.

Авторы благодарны и признательны всем тем, кто кто пришлёт замечания и пожелания по элоктронному адресу: [cool.araby@mail.ru](mailto:cool.araby@mail.ru) и gis\_71@mail.ru

# **Ввод и вывод данных, оператор присваивания: группа Begin**

Все входные и выходные данные в заданиях этой группы являются вещественными числами.

1. Дана сторона квадрата *a*. Найти его периметр

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  a, P: Real;  **begin**  ReadLn(a);  P := 4 \* a;  WriteLn('P = ', P);  **end**. | #include <iostream>  using namespace std;  int main()  {  double a,P;  cin >> a;  P=4\*a;  cout << "P -> " << P << endl;  return 0;  } | |
| Python | C# | |
| a=float(input("a->"))  P=4\*a  print(P) | static void Main(string[] args)  {  int a, p;  a = Convert.ToInt32(Console.ReadLine());  p = 4 \* a;  Console.Write(p);  Console.ReadKey();  } | |
| Java | | GO |
| public static void main(String[] args) {  int a,P;  Scanner pa=new Scanner(System.in);  a=pa.nextInt();  P=4\*a;  System.out.println(P);  } | | package main  import "fmt"  func main() {  fmt.Println("a=?")  var a int  fmt.Scanf("%d", &a)  var p=4\*a  fmt.Println("P=", p)  } |
| Kotlin | | |
| import java.util.Scanner fun main(args: Array<String>) {  val reader = Scanner(System.`in`)  var a=reader.nextFloat()  val p = 4\*a  println(p) } | | |
| VB.NET | | |
| Sub Main()  Dim a, p As Single  a = Convert.ToSingle(Console.ReadLine())  p = 4 \* a  Console.WriteLine(p)  End Sub | | |

1. Дан диаметр окружности *d*. Найти ее длину . В качестве значения использовать 3.14.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  d, L: Real;  **begin**  ReadLn(d);  L:=d\*pi;  WriteLn('L = ',L);  ReadLn;  **end**. | #include <iostream>  using namespace std;  int main(){  double d,L;  cin >> d;  L= d \* 3.14  cout<<"L="<<L<< endl;  return 0;  } | |
| Python | C# | |
| d= float(input("d:"))  L=3.14\*d  print(L) | static void Main(string[] args){  Double d, L, pi;  d = Convert.ToDouble(Console.ReadLine());  pi = Math.PI;  L = d \* pi;  Console.Write(L);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  double L,d;  Scanner ld=new Scanner(System.in);  d=ld.nextDouble();  L=Math.PI\*d;  System.out.println(L);  } | | package main  import "fmt"  func main() {  fmt.Println("d=?")  var d float64  fmt.Scanf("%v", &d)  var L=d\*3.14  fmt.Println("L=", L)  } |
| Kotlin | | |
| import java.util.Scanner fun main(args: Array<String>) {  val reader = Scanner(System.`in`)  var d=reader.nextFloat()  val L=Math.PI\*d  println(L) } | | |

1. Даны два ненулевых числа. Найти сумму, разность, произведение и частное их модулей.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a, b,s,m,z,t: Real;  **begin**  ReadLn(a, b);  s:= abs(a) + abs(b) ;  m:= abs(a) - abs(b) ;  z:= abs(a) \* abs(b) ;  t:= abs(a) / abs(b) ;  WriteLn(s,' ',m,' ',z, ' ',t);  **end**. | #include <iostream>  using namespace std;  int main()  {  double a, b,s,m,z,t;  cin >> a >> b;  s= abs(a) + abs(b) ;  m= abs(a) - abs(b) ;  z= abs(a) \* abs(b) ;  t= abs(a) / abs(b) ;  cout << "S= " <<s<< endl;  cout << "M=" <<m << endl;  cout << "Z=" << z<< endl;  cout << "T=" << t << endl;  return 0;  } |
| Python | C# |
| a=float(input("a:"))  b=float(input("b:"))  p=abs(a)+abs(b)  s=abs(a)-abs(b)  t=abs(a)\*abs(b)  v=abs(a)/abs(b)  print(p,"",s,"",v,"",t) | **public** static void **Main**(string[] args) {   **int** a, b, s, r, p, t;   a = Convert.**ToInt**(Console.**ReadLine**());   b = Convert.**ToInt**(Console.**ReadLine**());   s = Math.**Abs**(a) + Math.**Abs**(b);   r = Math.**Abs**(a) - Math.**Abs**(b);   p = Math.**Abs**(a) \* Math.**Abs**(b);   t = Math.**Abs**(a) / Math.**Abs**(b);   Console.**Write**("S={0}, R={1},P={2},t={2}",  s, r, p, t);   Console.**ReadKey**(**true**); } |
| Java | |
| public static void main(String[] args) {  int A, B,s,r,d,u;  Scanner rd=new Scanner (System.in);  a=rd.nextInt();  b=rd.nextInt();  s=Math.abs(a)+Math.abs(b);  r=Math.abs(a)-Math.abs(b);  d=Math.abs(a)/Math.abs(b);  u=Math.abs(a)\*Math.abs(b);  System.out.println(s+" "+r+" "+d+" "+u);  } | |
| Go | |
| package main  import ("fmt"  "math")  func main() {  fmt.Println("a, b=?")  var a,b float64  fmt.Scanf("%v %v", &a, &b)  var s=math.Abs(a)+math.Abs(b);  var r = math.Abs(a) – math.Abs(b);  var p = math.Abs(a) \* math.Abs(b);  var t = math.Abs(a) / math.Abs(b);  fmt.Println("S=",s,"R=",r, "P=",p,"T=",t)  } | |

1. Даны два круга с общим центром и радиусами и (> ). Найти площади этих кругов *S*1 и *S*2, а также площадь *S*3 кольца, внешний радиус которого равен , а внутренний радиус равен *R*2:

*S*1 = ·(*R*1)2,        *S*2 = ·(*R*2)2,        *S*3 = *S*1 – *S*2.

В качестве значения использовать 3.14.

| PascalABC.NET | C++ | | |
| --- | --- | --- | --- |
| **Var**  R1, R2, S1, S2, S3: Real;  **begin**  ReadLn(R1, R2);  S1:= pi \* Sqr(R1);  S2:= pi \* Sqr(R2);  S3 := S1 - S2;  WriteLn('S1=', S1,  ' S2=',S2,' S3=',  S3);  **end**. | #define PI 3.14159265  #include <iostream>  using namespace std;  int main(){  double R1, R2, S1, S2, S3;  cin >> R1 >> R2;  S1 = PI \* R1\*R1;  S2 = PI \* R2\*R2;  S3 = S1 - S2;  cout<<"S1 -> "<<S1<< "\nS2 -> " << S2 <<  "\nS3 ->" << S3 << endl;  system("pause");  return 0;  } | | |
| Python | | C# | |
| r1=float(input("r1:"))  r2=float(input("r2:"))  S1=3.14\*r1\*r1  S2=3.14\*r2\*r2  S=S1-S2  print(S) | | **public** static void **Main**(string[] args){         **double** r1,r2, s1,s2,S;  r1=Convert.**ToDouble**(Console.**ReadLine**());  r2=Convert.**ToDouble**(Console.**ReadLine**());   s1=Math.PI\*r1\*r1;   s2=Math.PI\*r2\*r2;   S=s1-s2;   Console.**WriteLine**(S);   Console.**ReadKey**(**true**);   } | |
| Java | | | GO |
| public static void main  (String[] args) {  Double S1,S2,S3,R1,R2;  Scanner ld=new  Scanner(System.in);  R1=ld.nextDouble();  R2=ld.nextDouble();  S1=Math.PI\*(R1\*R1);  S2=Math.PI\*(R2\*R2);  S3=S1-S2;  System.out.println(S1);  System.out.println(S2);  System.out.println(S3);  } | | | package main  import ("fmt"  "math")  func main() {  fmt.Println("Insigth R1 and R2!")  var R1,R2 float64  fmt.Scanf("%v %v", &R1, &R2)  var S1 = 3.14 \* math.Pow(R1,2);  var S2 = 3.14 \* math.Pow(R2,2);  var S3 = S1 - S2;  fmt.Println("S1=",S1,"\nS2=",S2, "\nS3=",S3)  } |

1. Найти расстояние между двумя точками с заданными координатами (*x*1, *y*1) и (*x*2, *y*2) на плоскости. Расстояние вычисляется по формуле

.

| PascalABC.NET | | | C++ |
| --- | --- | --- | --- |
| **Var**  x1,y1,x2,y2,S: Real;  **begin**  WriteLn('x1, y1');  RealLn(x1, y1);  WriteLn('x2, y2');  RealLn(x2, y2);  S=sqrt(sqr(x2-x1)+sqr(y2-  y1));  WriteLn('S = ', S);  **end**. | | | #define PI 3.14159265  #include <iostream>  using namespace std;  int main()  {  double x1, y1, x2, y2,s;  cout << "x1, y1 -> ";  cin >> x1 >> y1;  cout << "x2, y2 -> ";  cin >> x2 >> y2;  s=sqrt((y2 - y1) \* (x2 - x1)+y2-  y1)\*(y2-y1));  cout<<"S="<<s<<endl ;  system("pause");  return 0;  } |
| Python | C# | | |
| from math import sqrt  x1= float(input("x1:"))  x2= float(input("x2:"))  y1= float(input("y1:"))  y2= float(input("y2:"))  S=sqrt((x2-x1)\*\*2+(y2-y1)\*\*2)  print(S) | **public** static void **Main**(string[] args){   **double** x1, x2, y1, y2, s;  x1=Convert.**ToDouble**(Console.**ReadLine**());   y1=Convert.**ToDouble**(Console.**ReadLine**());   x2=Convert.**ToDouble**(Console.**ReadLine**());   y2=Convert.**ToDouble**(Console.**ReadLine**());  s=Math.**Sqrt**((x2 - x1) \* (x2 - x1) + (y2 - y1) \* (y2 - y1));   Console.**WriteLine**(s);   Console.**ReadKey**(**true**); } | | |
| Java | | Go | |
| public static void main(String[] args) {  double S,x1,x2,y1,y2;  Scanner cp=new Scanner  (System.in);  x1 = cp.nextDouble();  x2= cp.nextDouble();  y1= cp.nextDouble();  y2=cp.nextDouble();  S=Math.sqrt((x2-x1)\*  (x2-x1)-(y2-y1)\*(y2-y1));  System.out.println(S);  } | | package main  import ("fmt"  "math")  func main() {  fmt.Println("Insigth x1,x2 and y1,y2!")  var x1,x2,y1,y2 float64  fmt.Scanf("%v %v %v %v", &x1, &x2,  &y1, &y2)  var S=math.Sqrt((x2-x1)\*(x2-x1)-(y2-  y1)\*(y2-y1));  fmt.Println("S=",S)  } | |

1. Поменять местами содержимое переменных *A* и *B* и вывести новые значения *A* и *B*.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **var**  A, B, Temp: Real;  **begin**  WriteLn(' A, B');  RealLn(A, B);  C := A;  A := B;  B := C;  WriteLn('A =',A,' B=',B);  **end**. | #define PI 3.14159265  #include <iostream>  using namespace std;  int main(){  double A, B, C;  cout << "A, B:";  cin >> A >> B;  C = A;  A = B;  B = C;  cout<<"A="<<A<<"\nB="<<B<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("a:"))  b= int(input("b:"))  (a, b)=(b,a);  print(a," ", b) | **public** static void **Main**(string[] args) {   **int** a, b, c;   a = Convert.**ToInt**(Console.**ReadLine**());   b = Convert.**ToInt**(Console.**ReadLine**());   c=a;   a=b;   b=c;   Console.**WriteLine**(a+" "+b);   Console.**ReadKey**(**true**); } | |
| Java | | Go |
| public static void main(String[] args) {  int A, B,c;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  c=a;  a=b;  b=c;  System.out.println(a+" "+b);  } | | package main  import "fmt"  func main() {  var A,B float64  fmt.Scanf("%v %v", &A, &B)  A,B=B,A;  fmt.Println("A=",A,"\nB=",B)  } |

1. Дано число *A*. Вычислить *A*8, используя вспомогательную переменную и три операции умножения. Для этого последовательно находить *A*2, *A*4, *A*8. Вывести все найденные степени числа *A*.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A, Res: Real;  **begin**  ReadLn(A);  Res := A \* A;  WriteLn('A^2 = ', Res);  A := Res \* Res;  WriteLn('A^4 = ', A);  Res := A \* A;  WriteLn('A^8 = ', Res);  **end**. | #include <iostream>  using namespace std;  int main()  {  double A, Res;  cout << "A -> ";  cin >> A;  Res = A \* A;  cout <<"A^2->" << Res;  A = Res\*Res;  cout <<"A^4->" << A;  Res = A\*A;  cout <<"A^8->" << Res;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input(«a:»))  b=a\*a  print(b)  a=b\*b  b=a\*a  print(a," ", b) | **public** static void **Main**(string[] args){   **double** a,c;   a = Convert.**ToInt32**(Console.**ReadLine**());   c =a\*a;   Console.**WriteLine**(c);   a=c\*c;   c=a\*a;   Console.**WriteLine**(a+" "+c);   Console.**ReadKey**(**true**); } | |
| Java | | Go |
| public static void main(String[] args) {  double a,c;  Scanner sc=new Scanner(System.in);  a=sc.nextDouble();  c=a\*a;  System.out.print(c+" ");  a=c\*c;  c=a\*a;  System.out.println(a+" "+c);  } | | package main  import "fmt"  func main() {  var a,c float64  fmt.Scanf("%v", &a)  c=a\*a  fmt.Println(c,"\t")  a=c\*c  c=a\*a  fmt.Println(a,"\n",c)  } |

1. Дано значение угла в градусах (0° < < 360°). Определить значение этого же угла в радианах, учитывая, что 180° = радианов. В качестве значения использовать 3.14.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  alfa:Real;  **Begin**  ReadLn(alfa);  WriteLn(alfa\*pi/180,' радиан');  **End**. | #define PI 3.14159265  #include <iostream>  using namespace std;  int main(){  double alfa;  cout<<"ALFA->";  cin >> alfa;  cout<<(alfa\*PI)/180<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= float(input("a:"))  x=(a\*3.14)/180  print(x) | **public** static void **Main**(string[] args) {   **double** x,a;   a = Convert.**ToSingle**(Console.**ReadLine**());   x =(a\* 3.14) / 180 ;   Console.**WriteLine**(x);   Console.**ReadKey**(**true**); } | |
| Java | | Go |
| public static void main(String[] args) {  double a,x;  Scanner sc=new Scanner(System.in);  a=sc.nextDouble();  x=(Math.PI\*a)/180;  System.out.println(x);  } | | package main  import "fmt"  func main() {  fmt.Println("a!")  var a float64  fmt.Scanf("%v", &a)  var x=(3.14\*a)/180;  fmt.Println("x=",x)  } |

1. Скорость лодки в стоячей воде *V* км/ч, скорость течения реки *U* км/ч (*U* < *V*). Время движения лодки по озеру *T*1 ч, а по реке (против течения) — *T*2 ч. Определить путь *S*, пройденный лодкой (путь = время · скорость). Учесть, что при движении против течения скорость лодки уменьшается на величину скорости течения.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  V, U, T1, T2, S: Real;  **begin**  WriteLn('V,U,T1,T2:');  ReadLn(V, U, T1, T2);  S := T1 \* V + T2\*(V-U);  WriteLn('S= ', S);  **end**. | #include <iostream>  using namespace std;  int main()  {  double V, U, T1, T2, S;  cout<<"V,U,T1,T2->";  cin >> V >> U >> T1 >> T2;  S = T1\*V + T2\*(V - U);  cout<<S<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| u= float(input("u:"))  v= float(input("v:"))  t1= float(input("t1:"))  t2= float(input("t2:"))  S=t1\*v+(v-u)\*t2;  print(S) | **public** static void **Main**(string[] args)  {   **double** v, u, t1, t2, s;   v=Convert.**ToInt32**(Console.**ReadLine**());   u=Convert.**ToInt32**(Console.**ReadLine**());   t1=Convert.**ToInt32**(Console.**ReadLine**());   t2=Convert.**ToInt32**(Console.**ReadLine**());   s = v \* t1 + (v - u) \* t2;   Console.**WriteLine**(s);   Console.**ReadKey**(**true**); } | |
| Java | | Go |
| public static void main  (String[] args) {  double u,v,t1,t2,S;  Scanner sc=new Scanner  (System.in);  v=sc.nextDouble();  u=sc.nextDouble();  t1=sc.nextDouble();  t2=sc.nextDouble();;  S= t1\*v+(v-u)\*t2;  System.out.println(S);  } | | package main  import "fmt"  func main()  {  fmt.Println("u,v,t1,t2!")  var u,v,t1,t2 float64  fmt.Scanf("%v %v %v %v", &v,  &u,&t1,&t2)  var S=t1\*v+(v-u)\*t2  fmt.Println("S=",S)  } |

1. Найти корни *квадратного уравнения* , заданного своими коэффициентами *A*, *B*, *C* (коэффициент *A* не равен 0), если известно, что дискриминант уравнения положителен. Вывести сначала меньший, а затем больший из найденных корней. Корни квадратного уравнения находятся по формуле , где *D* — *дискриминант*, равный.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  a, b, c, x1, x2: real;  **begin**  writeln('A, B,c:');  readln(a, b, c);  d:=sqr(b) - 4 \* a \* c;  x1:=(-b-sqrt(d))/(2\*a);  x2:=(-b+sqrt(d))/(2\*a);  writeln('x1 = ', x1, '  x2 = ', x2);  **end**. | | #include <iostream>  using namespace std;  int main(){  double a, b, c, x1, x2,d;  cout << "enter a, b and c -> ";  cin>>a>>b>>c;  d = b\*b - 4\*a\*c;  x1 = (-b - sqrt(d))/(2\*a);  x2 = (-b + sqrt(d))/(2\*a);  cout<<"x1="<<x1<<"\nx2="<<x2<<endl;;  system("pause");  return 0;  } | |
| Python | C# | | |
| from math import sqrt  a= float(input("a:"))  b= float(input("b:"))  c= float(input("c:"))  d=b\*b-4\*a\*c  x1=(-b+sqrt(d))/2\*a  x2=(-b-sqrt(d))/2\*a  print(x1," ",x2) | **public** static void **Main**(string[] args){   **double** a, b, c, x1, x2, d;   a = Convert.**ToDouble**(Console.**ReadLine**());   b = Convert.**ToDouble**(Console.**ReadLine**());   c = Convert.**ToDouble**(Console.**ReadLine**());   d = b \* b - 4 \* a \* c;   x1 = (-b \* Math.**Sqrt**(d)) / (2 \* a);   x2 = (-b - Math.**Sqrt**(d)) / (2 \* a);   Console.**WriteLine**(x1 + " " + x2);   Console.**ReadKey**(**true**); } | | |
| Java | | | Go |
| public static void main(String[] args){  double A, B,c,d,x1,x2;  Scanner cp=new Scanner(System.in);  a=cp.nextDouble();  b=cp.nextDouble();  c=cp.nextDouble();  d=b\*b-4\*a\*c;  x1=(-b+Math.sqrt(d))/2\*a;  x2=(-b+Math.sqrt(d))/2\*a;  System.out.println(x1+" "+x2);  } | | | package main  import ("fmt"  "math"  )  func main() {  fmt.Println("a,b,c")  var a,b,c float64  fmt.Scanf("%v %v %v",&a,&b,&c)  var d = b\*b - 4\*a\*c;  var x1=(-b-math.Sqrt(d)) /(2\*a);  var x2=(-b+math.Sqrt(d))/(2\*a);  fmt.Println("x1=",x1,"\nx2=",x2)  } |
| Kotlin | | | |
| import java.util.Scanner fun main(args: Array<String>) {  val reader = Scanner(System.`in`)  var a=reader.nextDouble()  var b=reader.nextDouble()  var c=reader.nextDouble()  val d=b\*b-4\*a\*c  var x1=(-b+Math.sqrt(d))/2\*a  val x2=(-b-Math.sqrt(d))/2\*a  println("x1=$x1 x2=$x2") } | | | |

# **Целые числа: группа Integer**

Все входные и выходные данные в заданиях этой группы являются целыми числами. Все числа, для которых указано количество цифр (двузначное число, трехзначное число и т. д.), считаются положительными.

1. Дано расстояние *L* в сантиметрах. Используя операцию деления нацело, найти количество полных метров в нем (1 метр = 100 см).

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **var**  L, M: integer;  **begin**  read(L);  M := L **div** 100;  write(M);  **end**. | | #include <iostream>  using namespace std;  int main()  {  int L,M;  cin>>L;  M=L/100;  cout<<"M="<<M<<endl;  system("pause");  return 0;  } |
| Python | C# | |
| N= int(input("n:"))  M=int(L/100);  print(M); | static void Main(string[] args)  {  int L,M;  L= Convert.ToInt32(Console.ReadLine());  M = L/ 1000;  Console.Write(M);  Console.ReadKey();  } | |
| Kotlin | | |
| import java.util.Scanner fun main(args: Array<String>)  {  val reader = Scanner(System.`in`)  var L=reader.nextInt()  var m=L/100  println(m) } | | |
| VB.NET | | |
| Sub Main()  Dim L, m As Integer  L = Convert.ToInt32(Console.ReadLine())  m = L \ 100  Console.WriteLine(m)  Console.ReadKey()  End Sub | | |

1. Даны целые положительные числа и . На отрезке длины *A* размещено максимально возможное количество отрезков длины *B* (без наложений). Используя операцию деления нацело, найти количество отрезков *B*, размещенных на отрезке *A*.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A, B, Res: Integer;  **begin**  Write('A: ');  Readln(A);  Write('B: ');  Readln(B);  Res := A **div** B;  Writeln(Res);  **end**. | #include <iostream>  using namespace std;  int main(){  int A, B, Res;  cin>>A>>B;  Res=A/B;  cout<< Res<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("a: "))  b= int(input("b: "))  c=a//b  print(c) | static void Main(string[] args){  int a, b, y;  a= Convert.ToInt32(Console.  ReadLine());  b = Convert.ToInt32(Console.  ReadLine());  y = a/ b;  Console.WriteLine(y);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  int B,A,L;  Scanner sc=new Scanner (System.in);  A=sc.nextInt();  B=sc.nextInt();  L=A/B;  System.out.println(L);  } | | package main  import "fmt"  func main() {  fmt.Println("a,b")  var a,b int  fmt.Scanf("%d %d", &a,&b)  var L=a/b  fmt.Println("L=",L)  } |

1. Дано двузначное число. Вывести вначале его левую цифру (десятки), а затем — его правую цифру (единицы). Для нахождения десятков использовать операцию деления нацело, для нахождения единиц — операцию взятия остатка от деления.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  a, r, L: integer;  **begin**  WriteLn('a');  ReadLn(a);  r := a **div** 10;  L := a **mod** 10;  WriteLn('r=', r);  WriteLn('L=', L);  **end**. | #include <iostream>  using namespace std;  int main(){  int A, B,c;  cout<<"n=: ";  cin>>a;  b = a / 10;  c = a % 10;  cout<<b<<" "<<c;  system("pause");  return 0;  } | |
| Python | C# | |
| n= int(input("n:"))  a=n//10  b=n%10  print(a,' ',b) | static void Main(string[] args){  int L, m,c;  L = Convert.ToInt32(Console.ReadLine());  m = L / 10;  c = L % 10;  Console.WriteLine(m+" "+c);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  int A, B,N;  Scanner sc=new Scanner(System.in);  N=sc.nextInt();  a=N/10;  b=N%10;  System.out.println(a+” “+b);  } | | package main  import "fmt"  func main() {  fmt.Println("a->!")  var n int  fmt.Scanf("%d", &n)  var a=n/10;  var b=n%10; fmt.Println("a=",a,"\nb=",b)  } |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** n=reader.nextInt()  **var** a=n/10  **var** b=n%10  *println*(**"a=$**a **b=$**b**"**) } | | |
| VB.NET | | |
| Sub Main()  Dim n, a, b As Integer  n = Convert.ToInt32(Console.ReadLine())  a = n \ 10  b = n Mod 10  Console.WriteLine("{0} {1}", a, b)  Console.ReadKey()  End Sub | | |

1. Дано трехзначное число. Используя одну операцию деления нацело, вывести первую цифру данного числа (сотни).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  a, r: integer;  **begin**  Writeln('a');  readln(a);  r := a **div** 100;  Writeln('r=', r);  **end**. | | #include <iostream>  using namespace std;  int main(){  int a;  cout<<"a: ";  cin>>a;  a = a / 100;  cout<<a<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("n:"))  a=n//100  print(a) | static void Main(string[] args){  int l, m;  l=Convert.ToInt32(Console.ReadLine());  m = l / 100;  Console.WriteLine(m);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int a,N;  Scanner sc=new Scanner(System.in);  N=sc.nextInt();  a=N/100;  System.out.println(a);  } | | | package main  import "fmt"  func main() {  fmt.Println("a->")  var n int  fmt.Scanf("%d", &n)  var a=n/100;  fmt.Println("Сади=",a)  } |

1. Дано трехзначное число. Вывести число, полученное при прочтении исходного числа справа налево.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  x, a, b, c, p: integer;  **begin**  Writeln('a');  readln(x);  a := x **div** 100;  b:=(x **div** 10) **mod** 10;  c := x **mod** 10;  p:=c\*100+b\*10+a;  Writeln('p =', p);  **end**. | #include <iostream>  using namespace std;  int main(){  int A, B,c,d,f;  cout<<"n:";  cin>>a;  b = a / 100;  c = a % 100;  d = c % 10;  f = c / 10;  cout<<d<<" "<<f<<" "<<b<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| n= int(input("n: "))  a=n//100  b=(n%100)//10  c=n%10  S=c\*100+b\*10+a  print(S) | static void Main(string[] args){  int a, b,c,x,m,n ;  n = Convert.ToInt32(Console.ReadLine());  a = n / 100;  x = n % 100;  b = x / 10;  c = x % 10;  m = 100 \* c + 10 \* b + a;  Console.WriteLine(m);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  int A, B,c,N,M;  Scanner sc=new Scanner(System.in);  N=sc.nextInt();  a=N/100;  b=(N%100)/10;  c=(N%10)%10;  M=100\*c+10\*b+a;  System.out.println(M);  } | | package main  import "fmt"  func main() {  fmt.Println("n->")  var n int  fmt.Scanf("%d", &n)  var a=n/100;  var b=(n%100)/10;  var c=(n%10)%10;  var M=100\*c+10\*b+a;  fmt.Println("M=",M)  } |

1. С начала суток прошло *N* секунд (*N* — целое). Найти количество полных минут, прошедших с начала суток.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  s, m: integer;  **begin**  readln(s);  m := s **div** 60;  writeln('m =', m)  **end**. | #include <iostream>  using namespace std;  int main(){  int t;cin>>t;  t = t / 60;  cout<<"m="<<t;  system("pause");  return 0;  } | |
| Python | C# | |
| s= int(input("s:"))  m=s//60  print(m) | static void Main(string[] args){  int m,n ;  n = Convert.ToInt32(Console.ReadLine());  m = n / 60;  Console.WriteLine(m);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  int Nc,M;  Scanner sc=new Scanner(System.in);  Nc=sc.nextInt();  M=Nc/60;  System.out.println(M);  } | | package main  import "fmt"  func main()  {  fmt.Println("n>999)")  var ns,M int  fmt.Scanf("%d", &ns)  M=ns/60;  fmt.Println("M=",M)  } |

1. Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, …, 6 — суббота. Дано целое число *K*, лежащее в диапазоне 1–365. Определить номер дня недели для *K*-го дня года, если известно, что в этом году 1 января было понедельником.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  x, d: integer;  **begin**  readln(x);  d := x **mod** 7;  writeln(d);  **end**. | #include <iostream>  using namespace std;  int main()  {  int d; cin>>d;  d = d % 7;  cout<<d<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| s= int(input("s:"))  m=s%7  print(m) | static void Main(string[] args){  int y,k ;  k=Convert.ToInt32(Console.ReadLine());  y = k % 7;  Console.WriteLine(y);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  int k,L;  Scanner lk=new Scanner(System.in);  k=lk.nextInt();  L=k%7;  System.out.println(L);  } | | package main  import "fmt"  func main() {  fmt.Println("k->")  var k,L int  fmt.Scanf("%d", &k)  L=k%7; fmt.Println(L)  } |

1. Даны целые положительные числа *A*, *B*, *C*. На прямоугольнике размера *A* ≠ *B* размещено максимально возможное количество квадратов со стороной *C* (без наложений). Найти количество квадратов, размещенных на прямоугольнике, а также площадь незанятой части прямоугольника.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  a, b, c, s, p: integer;  Resulr: real;  **begin**  readln(a, b, c);  p:=(a **div** c)\*(b **div** c);  s := b \* (a **mod** c) + (b **mod** c)\*(a-a **mod** c);  writeln(s, ' ', p);  **end**. | | #include <iostream>  using namespace std;  int main(){  int A, B,c,p,s;  cin>>a>>b>>c;  p = (a/c) \* (b/c);  s = b\*(a % c) + (b % c)\*(a-a%c);  cout<<p<<" "<<s<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input(«a:»))  b= int(input(«b:»))  c= int(input(«c:»))  p = (a//c) \* (b//c);  s = b\*(a % c) + (b % c)\*(a-a%c);  print(p,' ',s ) | static void Main(string[] args){  int A, B,c,s,p;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  c = Convert.ToInt32(Console.ReadLine());  p = (a/c) \* (b/c);  s = b\*(a % c) + (b % c)\*(a-a%c);  Console.WriteLine(p + " " + s);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args){  int A, B,c,p,s;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  c=sc.nextInt();  p = (a/c) \* (b/c);  s = b\*(a % c) + (b % c)\*(a-a%c);  System.out.println(p+” “+s);  } | | | package main  import "fmt"  func main() {  fmt.Println("a,b,c->")  var a,b,c,s,p int  fmt.Scanf("%d %d %d",&a,&b,&c)  p = (a/c) \* (b/c);  s=b\*(a % c)+(b%c)\*(a-a%c); fmt.Println("P=",p,"\nS=",s)  } |

**Логические выражения: группа Boolean**

Во всех заданиях данной группы требуется вывести логическое значение True, если приведенное высказывание для предложенных исходных данных является истинным, и значение False в противном случае. Все числа, для которых указано количество цифр (двузначное число, трехзначное число и т. д.), считаются целыми положительными.

1. Дано целое число *A*. Проверить истинность высказывания: «Число *A* является положительным».

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A: Integer;  **begin**  ReadLn(A);  WriteLn(A > 0);  **end**. | #include <iostream>  using namespace std;  int main()  {  int A;  cin >> A;  cout << (A > 0);  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("a: "))  y=a>0  print(y) | static void Main(string[] args){  int a;  bool y;  a = Convert.ToInt32(Console.ReadLine());  y = a > 0;  Console.WriteLine(y);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  boolean y;  int A ;  Scanner sc=new Scanner(System.in);  A=sc.nextInt();  y=A>0;  System.out.println(y);  } | | package main  import "fmt"  func main() {  fmt.Println("a=?")  var a int  var y bool  fmt.Scanf("%d", &a)  y=a>0  fmt.Println("y=", y)  } |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a=reader.nextInt()  **var** y=a>0  *println*(**"y=$**y**"**) } | | |
| VB.NET | | |
| Sub Main()  Dim a As Integer  Dim y As Boolean  a = Convert.ToInt32(Console.ReadLine())  y = a > 0  Console.WriteLine(y)  Console.ReadKey()  End Sub | | |

1. Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Справедливы неравенства *A* > 0 или *B* < –2».

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A, B: Integer;  y: Boolean;  **begin**  ReadLn(A, B);  y:=(A>2) **And** (B<=3);  WriteLn(y);  **end**. | #include <iostream>  using namespace std;  int main(){  int A, B;  cin >> A >> B;  cout << ((A > 2) && (B <= 3));  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("a: "))  b= int(input("b: "))  y=(a>2) and (b<=3)  print(y) | static void Main(string[] args){  bool y;  int a, b;  a= Convert.ToInt32(Console.ReadLine());  b= Convert.ToInt32(Console.ReadLine());  y = (a > 2) && (b <= 3);  Console.WriteLine(y);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  boolean y;  int A, B ;  Scanner sc=new Scanner(System.in);  A=sc.nextInt();  B=sc.nextInt();  y=(A>2)&&(B<=3);  System.out.println(y);  } | | package main  import "fmt"  func main() {  fmt.Println("a=?")  var a,b int  var y bool  fmt.Scanf("%d %d",&a,&b)  y=(a>2) && (b<=3)  fmt.Println("y=", y)  } |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a=reader.nextInt()  **var** b=reader.nextInt()  **var** y=(a>2) && (b<=3)  *println*(**"y=$**y**"**) } | | |

1. Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Каждое из чисел *A* и *B* нечетное».

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A, B: Integer;  **begin**  ReadLn(A, B);  WriteLn(Odd(A) **And** Odd(B));  **end**. | #include <iostream>  using namespace std;  int main()  {  int A, B;  cin >> A >> B;  bool y=((A % 2 != 0)&&(B % 2 != 0));  cout << y;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input(«a:»))  b=int(input("b: "))  y=(a%2)\*(b%2)==1  print(y) | static void Main(string[] args) {  int a, b;  bool y;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  y = ((a % 2 == 0)) && ((a % 2 == 0));  Console.WriteLine(y);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  boolean y;  int A, B ;  Scanner sc=new Scanner(System.in);  A=sc.nextInt();  B=sc.nextInt();  y=((A%2==1)&&(B%2==1));  System.out.println(y);  } | | package main  import ("fmt"  )  func main() {  fmt.Println("а, b")  var a,b int  var y bool  fmt.Scanf("%d %d",&a,&b)  y=((a%2==1)&&(b%2==1));  fmt.Println("Y=",y)  } |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a=reader.nextInt()  **var** b=reader.nextInt()  **var** y=(a%2==0) && (b%2==0)  *println*(**"y=$**y**"**) } | | |

1. Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Ровно одно из чисел *A* и *B* нечетное».

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B: Integer;  **begin**  ReadLn(A, B);  WriteLn(Odd(A) **Xor** Odd(B));  **end**. | | #include <iostream>  using namespace std;  int main(){  int a, b;  bool y;  cin >> a >> b;  y=((b%2==1) &&(a%2==0))||  ((a%2==0)&&(b%2==1));  cout << y<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("a: "))  b= int(input("b: "))  y=(a%2)+(b%2)==1  print(y) | static void Main(string[] args){  int a, b;  bool y;  a=Convert.ToInt32(Console.ReadLine());  b=Convert.ToInt32(Console.ReadLine());  y = ((b % 2 == 1) &&(a % 2 == 0))||  ((a%2==0)&&(b%2==1));  Console.WriteLine(y);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args) {  boolean y;  int A, B;  Scanner sc=new Scanner (System.in);  A=sc.nextInt();  B=sc.nextInt();  y=((A%2==1)&&(B%2==0))||((A%2==0)&&  (B%2==1));  System.out.println(y);  } | | | package main  import "fmt"  func main() {  fmt.Println("A, B")  var A,B int  var y bool  fmt.Scanf("%d %d",&A,&B)  y=((A%2==1)&&(B%2==0))||  ((A%2==0)&&(B%2==1));  fmt.Println("Y=",y)  } |

1. Проверить истинность высказывания: «Среди трех данных целых чисел есть хотя бы одна пара совпадающих».

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B, C: Integer;  y: Boolean;  **begin**  ReadLn(A, B, C);  y:=(A=B) **Or** (B=C) **Or** (A=C);  WriteLn(y);  **end**. | | #include <iostream>  using namespace std;  int main()  {  int A, B, C;  cin >> A >> B >> C;  bool y=((A == B) || (B == C) ||  (A == C));  cout <<y<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("a: "))  b = int(input("b:"))  c = int(input("c:"))  y=(a==b) or (a==c) or  (b==c)  print(y) | static void Main(string[] args)  {  int a, b, c;  bool y;  a= Convert.ToInt32(Console.ReadLine());  b= Convert.ToInt32(Console.ReadLine());  c= Convert.ToInt32(Console.ReadLine());  y=(a==b)||(a==c) || (b == c);  Console.WriteLine(y);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args){  boolean y;  int A, B,c;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  c=sc.nextInt();  y=(a==b)||(b==c)||(c==a);  System.out.println(y);  } | | | package main  import "fmt"  func main() {  var a,b,c int  var y bool  fmt.Scanf("%d %d %d",  &a,&b,&c)  y=(a==b)||(b==c)||(c==a);  fmt.Println("Y=",y)  } |

1. Даны числа . Проверить истинность высказывания: «Точка с координатами лежит во второй координатной четверти».

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **var**  x, y: Real;  z:Boolean;  **begin**  ReadLn(x, y);  z:=(x<=0) **And** (y>=0);  WriteLn(z);  **end**. | #include <iostream>  using namespace std;  int main(){  double x, y;  cin >> x >> y;  cout<<((x <= 0) && (y >= 0));  system("pause");  return 0;  } | |
| Python | C# | |
| x= int(input(«x:»))  y = float(input(":"))  z=(x<0) and (y>0)  print(z) | static void Main(string[] args){  bool L;  int x,y;  x = Convert.ToInt32(Console.ReadLine());  y = Convert.ToInt32(Console.ReadLine());  L = (x < 0) && (y > 0);  Console.WriteLine(L);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  boolean L;  Double x,y;  Scanner sc=new Scanner (System.in);  x=sc.nextDouble();  y=sc.nextDouble();  L=(x<0)&&(y>0);  System.out.println(L);  } | | package main  import "fmt"  func main() {  fmt.Println("x,y->")  var x,y int  var L bool  fmt.Scanf("%d %d",&x,&y)  L=(x<0)&&(y>0);  fmt.Println("L=",L) |

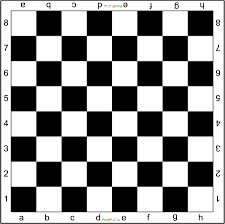
1. Даны числа *x*, *y*, *x*1, *y*1, *x*2, *y*2. Проверить истинность высказывания: «Точка с координатами (*x*, *y*) лежит внутри прямоугольника, левая верхняя вершина которого имеет координаты (*x*1, *y*1), правая нижняя — (*x*2, *y*2), а стороны параллельны координатным осям».

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **var**  x, y, x1, y1, x2, y2: Real;  z: Boolean;  **begin**  WriteLn('(x,y)');  ReadLn(x, y);  WriteLn('(x1,y1)');  ReadLn(x1, y1);  WriteLn('(x2,y2)');  ReadLn(x2, y2);  z:=(y<y1) **And** (y>y2) **And**  (x > x1) **And** (x < x2);  WriteLn(z);  **end**. | | #include <iostream>  using namespace std;  int main(){  float x, y, x1, y1, x2, y2;  cout << "(x, y): ";  cin >> x >> y;  cout << "(x1, y1): ";  cin >> x1 >> y1;  cout << "(x2, y2): ";  cin >> x2 >> y2;  bool w=((y < y1) && (y > y2) &&  (x > x1) && (x < x2));  cout <<w ;  system("pause");  return 0;  } | |
| Python | C# | | |
| x= float(input("->"))  y = float(input("->"))  x1= float(input("->"))  y1 = float(input("->"))  x2= float(input("->"))  x2 = float(input("->"))  z = ((x1 < x) and (x<x2) and (y2<y) and (y<y1))  print(z) | static void Main(string[] args){  bool z;  double x, x1, x2, y, y1, y2;  x=Convert.ToDouble(Console.ReadLine());  x1=Convert.ToDouble(Console.ReadLine());  x2=Convert.ToDouble(Console.ReadLine());  y=Convert.ToDouble(Console.ReadLine());  y1=Convert.ToDouble(Console.ReadLine());  y2=Convert.ToDouble(Console.ReadLine());  z=((x1<x) && (x<x2) && (y2<y) && (y<y1));  Console.Write(z);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  boolean L;  int x,x1,x2,y,y1,y2;  Scanner sc=new Scanner(System.in);  x=sc.nextInt();  x1=sc.nextInt();  x2=sc.nextInt();  y=sc.nextInt();  y1=sc.nextInt();  y2=sc.nextInt();  L=(x1<x)&&(x<x2)&&(y1<y)&&  (y<y2);  System.out.println(L);  } | | | package main  import "fmt"  func main() {  fmt.Println("x,y,x1,x2,y1,y2")  var x1,x2,y1,y2,x,y int  var L bool  fmt.Scanf("%d %d %d %d %d  %d",&x1,&x2,&y1,&y2,&x,&y)  L=(x1<x)&&(x<x2)&&(y1<y)&&  (y<y2);  fmt.Println("L=",L)  } |

1. Даны целые числа *a*, *b*, *c*, являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами *a*, *b*, *c* является равнобедренным».

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  x,y,x1,y1,x2,y2:Real;  z: Boolean;  **begin**  WriteLn('(x,y)');  ReadLn(x, y);  WriteLn('(x1,y1)');  ReadLn(x1, y1);  WriteLn('(x2,y2)');  ReadLn(x2, y2);  z :=(y<y1) **And** (y > y2)  **And** (x>x1) **And** (x<x2);  WriteLn(z);  **end**. | #include <iostream>  using namespace std;  int main()  {  int a, b, c;  cin >> a >> b >> c;  bool s=((a==b)||(b==c)||(c==b));  cout << s;  system("pause");  return 0;  } | |
| Python | C# | |
| a= float(input("->"))  b = float(input("->"))  c= float(input("->"))  z = (a==b) or (a==c) or  (b==c)  print(z) | static void Main(string[] args){  bool y;  int a, b, c;  a=Convert.ToInt32(Console.ReadLine());  b=Convert.ToInt32(Console.ReadLine());  c=Convert.ToInt32(Console.ReadLine());  y=((a==b)||(b==c)&&(c==a));  Console.Write(y);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  boolean L;  int a,b,c;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  c=sc.nextInt();  L=(a==b) ||(b==c) ||(c==a);  System.out.println(L);  } | | package main  import "fmt"  func main() {  fmt.Println("a,b,c->")  var a,b,c int  var L bool  fmt.Scanf("%d %d %d",&a,&b,&c)  L=(a==b) ||(b==c) ||(c==a);  fmt.Println("L=",L)  } |

1. Даны координаты поля шахматной доски *x*, *y* (целые числа, лежащие в диапазоне 1–8). Учитывая, что левое нижнее поле доски (1, 1) является черным, проверить истинность высказывания: «Данное поле является белым».



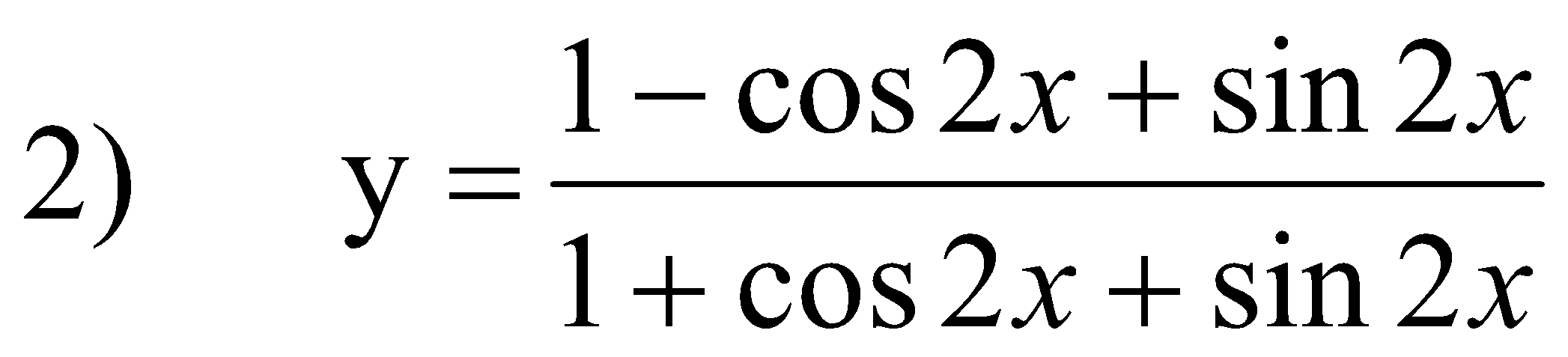
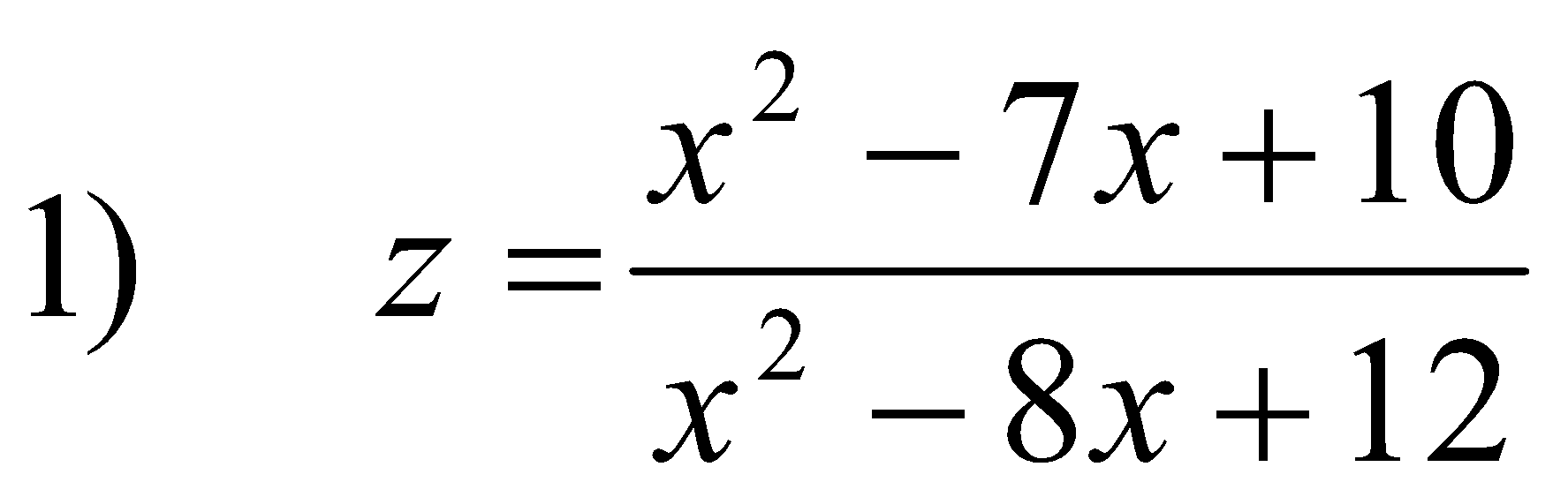
| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  **var**  x, y: Integer;  z:Boolean;  **begin**  ReadLn(x, y);  z:=**not** ((y + x) **Mod** 2 = 0);  WriteLn(z);  **end**. | | #include <iostream>  using namespace std;  int main(){  int x, y;  cin >> x >> y;  cout << !((x + y) % 2 == 0);  system("pause");  return 0;  } | |
| Python | C# | | |
| x= int(input("->"))  y = int(input("->"))  z=(x+y)%2==1  print(z) | static void Main(string[] args){  bool k;  int x, y;  x= Convert.ToInt32(Console.ReadLine());  y= Convert.ToInt32(Console.ReadLine());  k = (x+y) % 2 == 1;  Console.Write(k);  Console.ReadKey()  } | | |
| Java | | | Go |
| public static void main(String[] args) {  boolean L;  int x,y;  Scanner sc=new Scanner(System.in);  x=sc.nextInt();  y=sc.nextInt();  L=(x+y) % 2 == 1;  System.out.println(L);  } | | | package main  import "fmt"  func main() {  fmt.Println("x,y")  var x,y int  var L bool  fmt.Scanf("%d %d",&x,&y)  L=(x+y) % 2 == 1;  fmt.Println("L=",L)  } |

1. Даны координаты двух различных полей шахматной доски *x*1, *y*1,   
   *x*2, *y*2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Ладья за один ход может перейти с одного поля на другое».

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **var**  x1, y1, x2, y2: Integer;  z: Boolean;  **begin**  ReadLn(x1, y1);  ReadLn(x2, y2);  z:=(x1 = x2) **Or** (y1 = y2);  WriteLn(z);  **end**. | | #include <iostream>  using namespace std;  int main(){  int x1, x2, y1, y2;  cin >> x1 >> y1>> x2 >> y2;  cout<<((x1==x2) || (y1==y2));  system("pause");  return 0;  } | |
| Python | C# | | |
| x1= int(input("->"))  y1 = int(input("->"))  x2= int(input("->"))  y2 = int(input("->"))  z = (x1 == x2) or (y1 == y2)  print(z) | static void Main(string[] args){  int x1, x2, y1, y2;  bool z;  x1=Convert.ToInt32(Console.ReadLine());  y1=Convert.ToInt32(Console.ReadLine());  x2=Convert.ToInt32(Console.ReadLine());  y2=Convert.ToInt32(Console.ReadLine());  z = (x1 == x2) || (y1 == y2);  Console.WriteLine(z);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args){  boolean L;  int x1, x2, y1, y2;  Scanner sc=new Scanner(System.in);  x1=sc.nextInt();  y1=sc.nextInt();  x2=sc.nextInt();  y2=sc.nextInt();  L=(x1 == x2) || (y1 == y2);  System.out.println(L);  } | | | package main  import "fmt"  func main(){  fmt.Println("x1,x2,y1,y2")  var x1,x2,y1,y2 int  var L bool  fmt.Scanf("%d %d %d %d",&x1,  &x2,&y1,&y2)  L=(x1 == x2) || (y1 == y2);  fmt.Println("L=",L)  } |

## **Задачи для самостоятельной работы**

**Вариант** 1

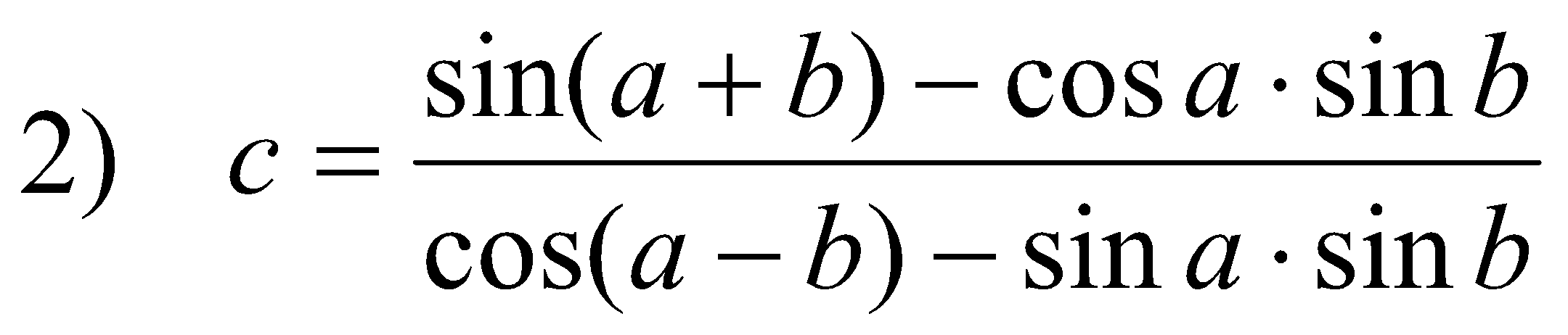
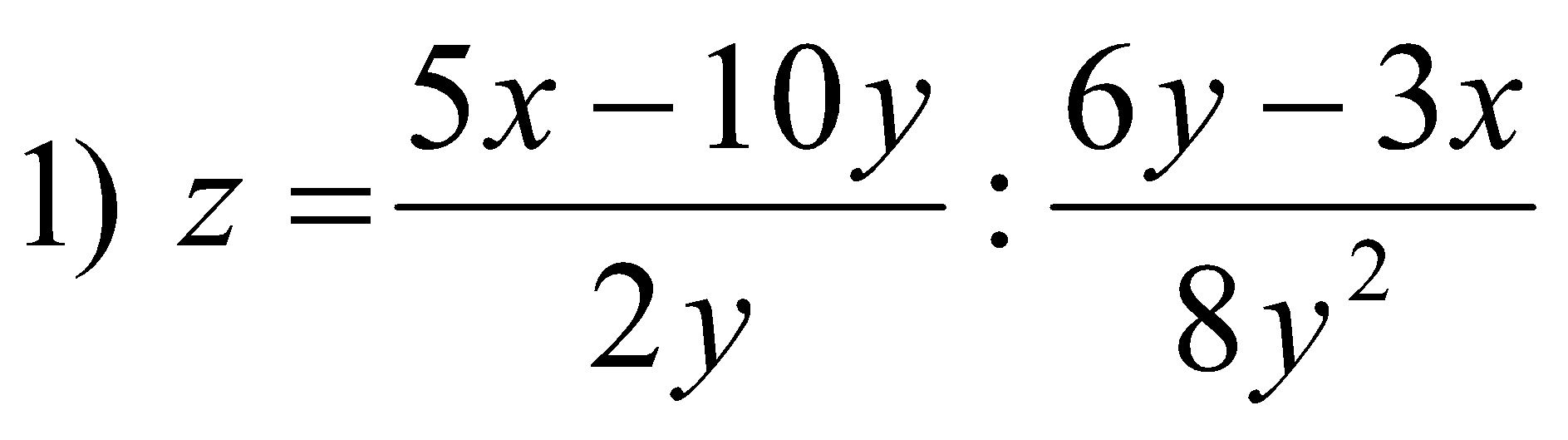


3) Заданы координаты трех вершин треугольника (x1,y1), (x2,y2), (x3,y3).Найти его периметр и площадь.

4) Даны целые положительные числа *A* и *B* (*A* > *B*). На отрезке длины *A* размещено максимально возможное количество отрезков длины *B* (без наложений). Используя операцию деления нацело, найти количество отрезков *B*, размещенных на отрезке *A*.

5)Даны координаты двух различных полей шахматной доски *x*1, *y*1,   
*x*2, *y*2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Конь за один ход может перейти с одного поля на другое».

**Вариант 2**

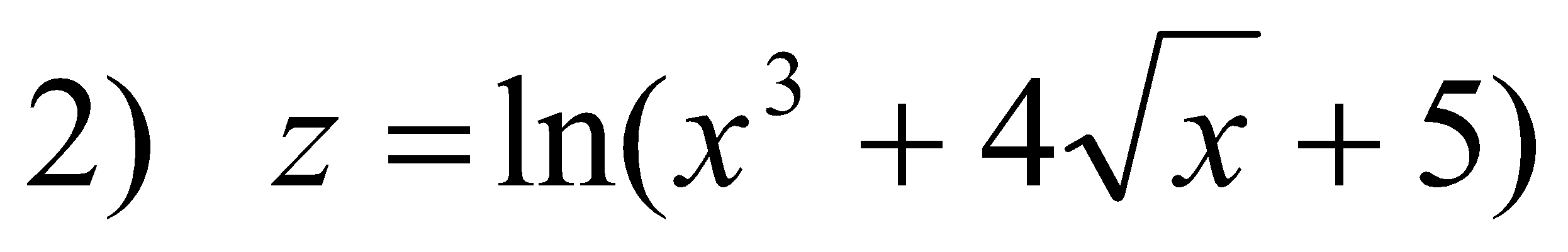
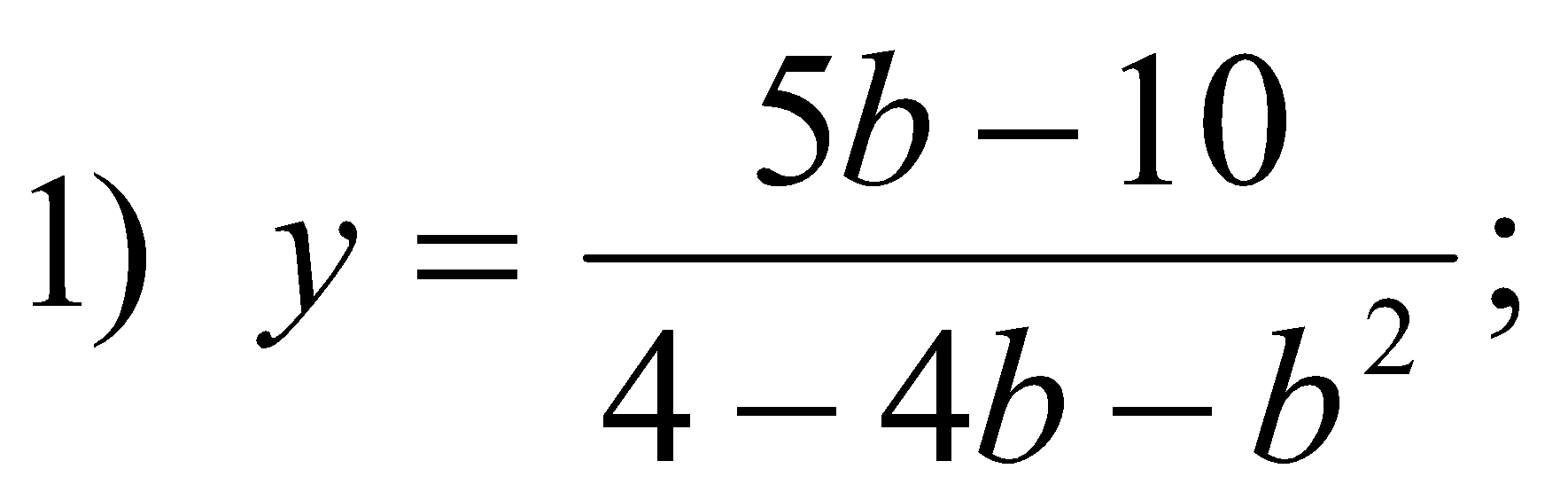


3) Дана сторона равностороннего треугольника. Найти площадь этого треугольника, его высоты, радиусы вписанной и описанной окружностей.

4) Даны целые положительные числа *A* и *B* (*A* > *B*). На отрезке длины *A* размещено максимально возможное количество отрезков длины *B* (без наложений). Используя операцию взятия остатка от деления нацело, найти длину незанятой части отрезка *A*.

5) Даны координаты двух различных полей шахматной доски *x*1, *y*1,   
*x*2, *y*2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Ферзь за один ход может перейти с одного поля на другое».

**Вариант 3**

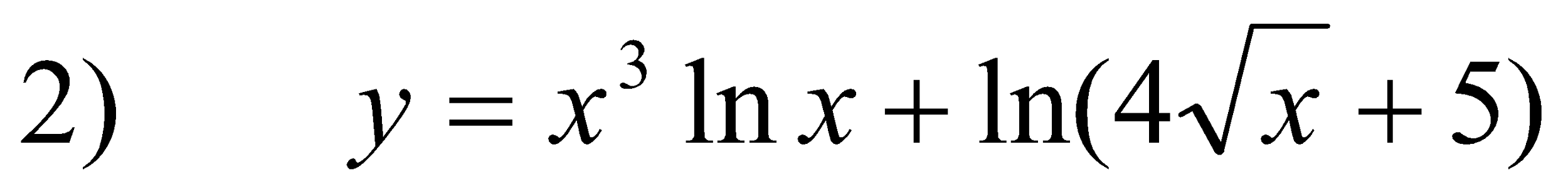
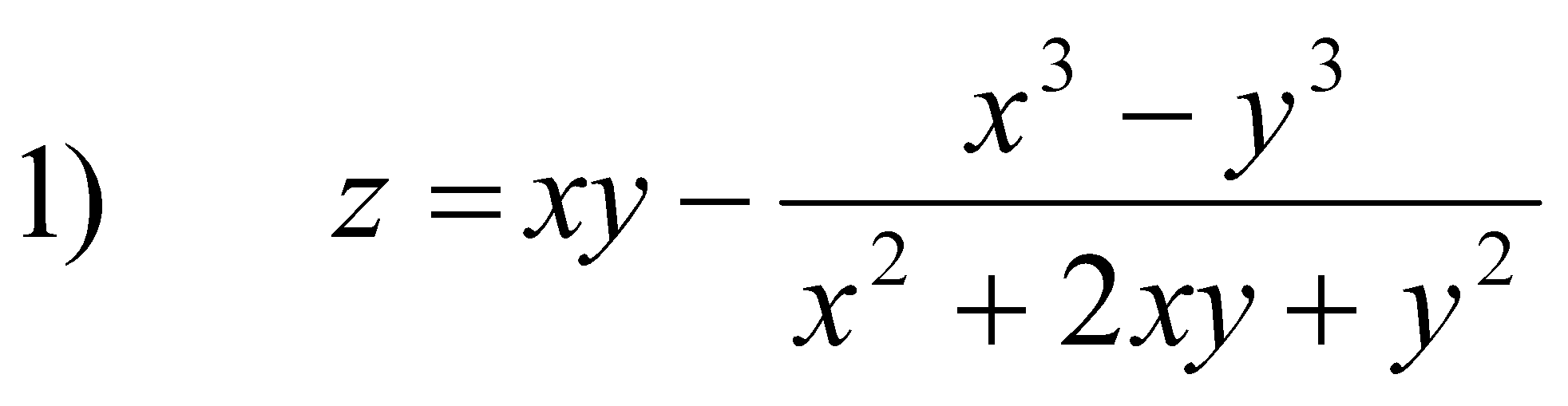


3) Найти сумму членов арифметической прогрессии, если известны ее первый член, знаменатель и число членов прогрессии.

4) Дано двузначное число. Вывести вначале его левую цифру (десятки), а затем — его правую цифру (единицы). Для нахождения десятков использовать операцию деления нацело, для нахождения единиц — операцию взятия остатка от деления.

5)Даны координаты двух различных полей шахматной доски *x*1, *y*1,   
*x*2, *y*2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Слон за один ход может перейти с одного поля на другое».

**Вариант 4**

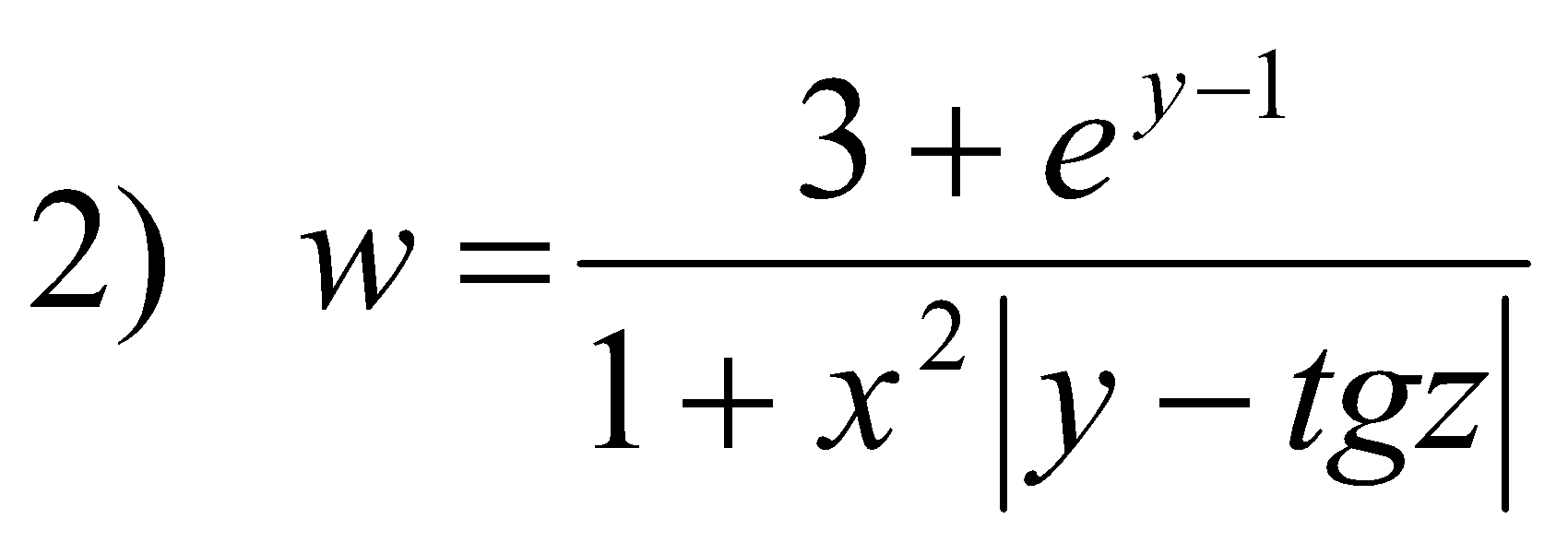
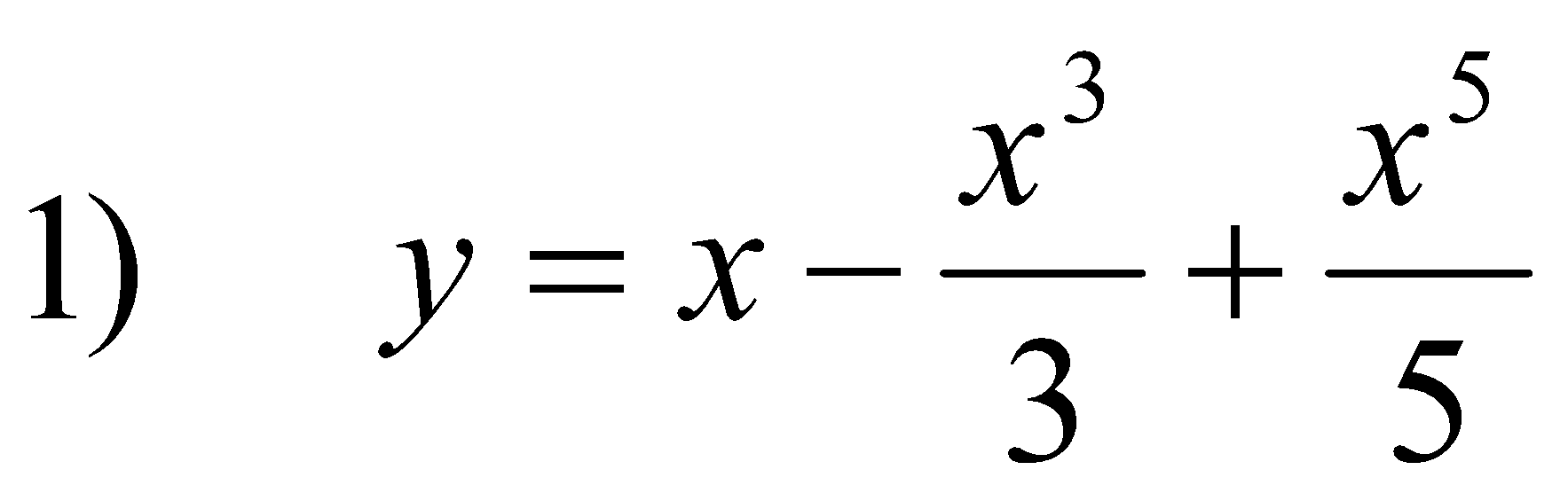


3) Вычислить высоты треугольника со сторонами a, b, c.

4) Дано двузначное число. Найти сумму и произведение его цифр.

5)Даны координаты двух различных полей шахматной доски *x*1, *y*1,   
*x*2, *y*2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Король за один ход может перейти с одного поля на другое».

**Вариант 5**

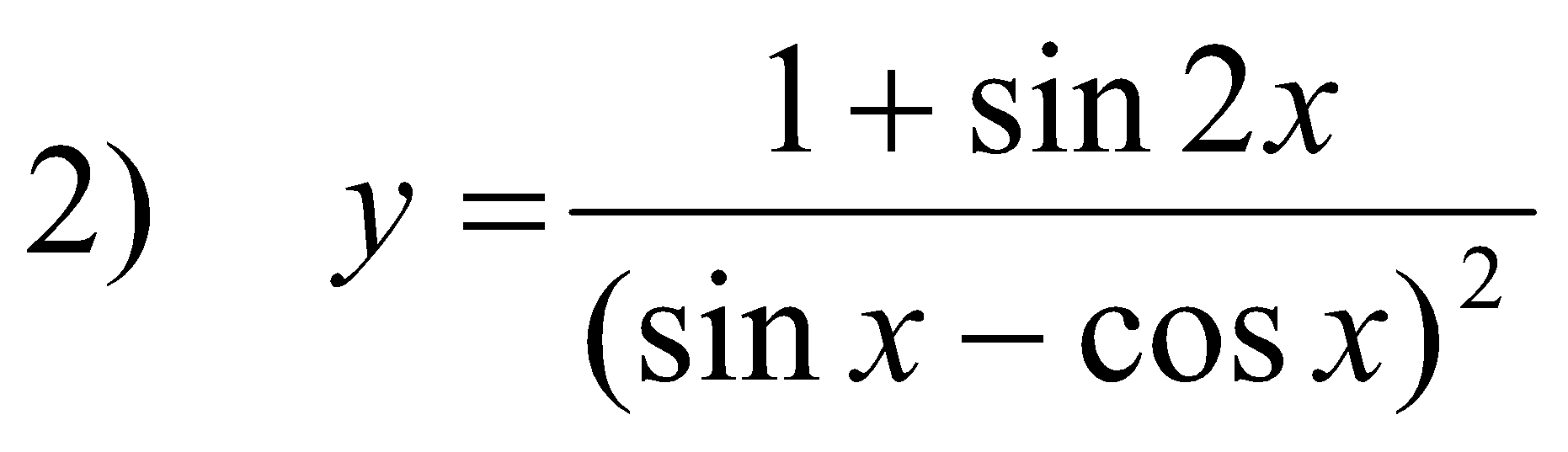
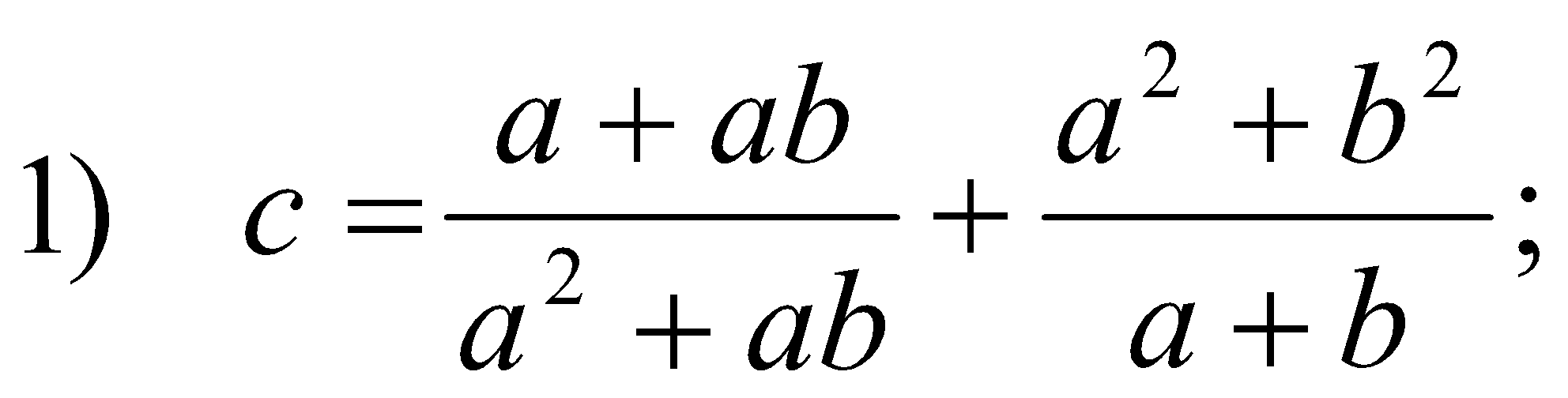


3)Даны два числа. Найти среднее арифметическое кубов этих чисел и среднее геометрическое модулей этих чисел.

4) Дано двузначное число. Вывести число, полученное при перестановке цифр исходного числа.

5)Даны координаты двух различных полей шахматной доски *x*1, *y*1,   
*x*2, *y*2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Ладья за один ход может перейти с одного поля на другое».

**Вариант 6**

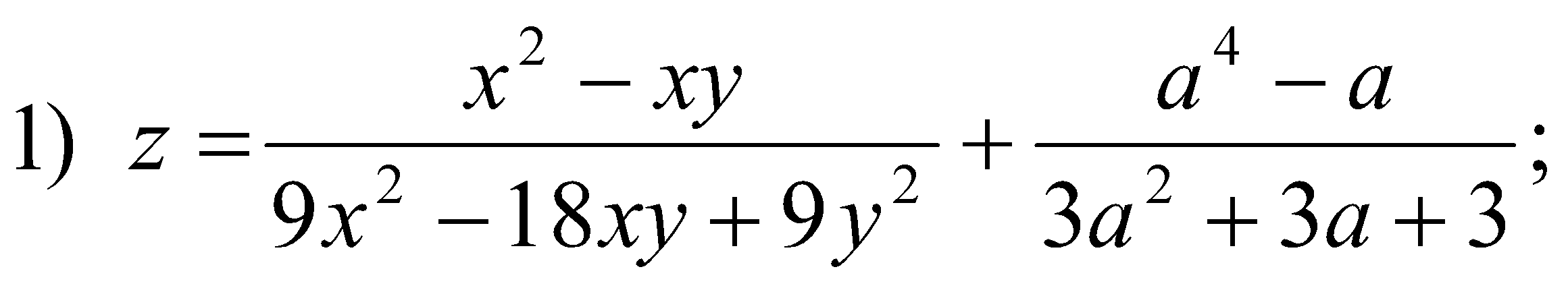


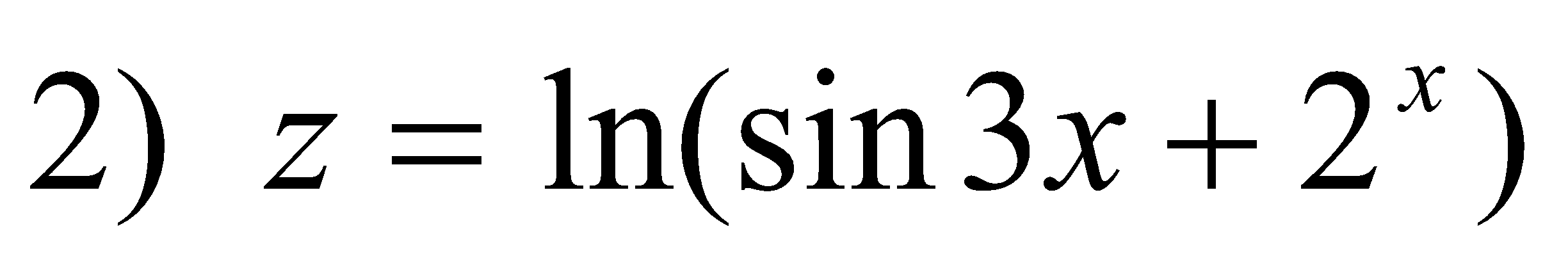
3) Известна длина окружности. Найти площадь круга, ограниченного этой окружностью.

4) Дано трехзначное число. Используя одну операцию деления нацело, вывести первую цифру данного числа (сотни).

5)Даны координаты двух различных полей шахматной доски *x*1, *y*1,   
*x*2, *y*2 (целые числа, лежащие в диапазоне 1–8). Проверить истинность высказывания: «Король за один ход может перейти с одного поля на другое».

**Вариант 7**

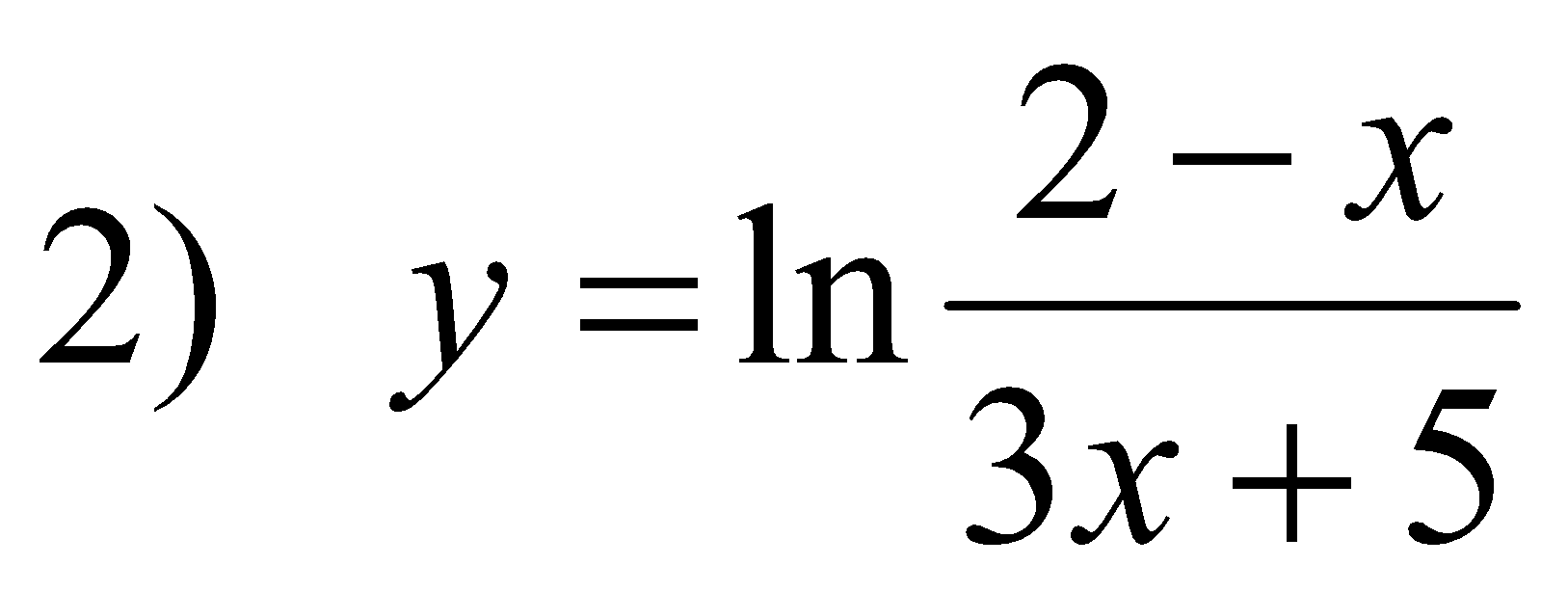
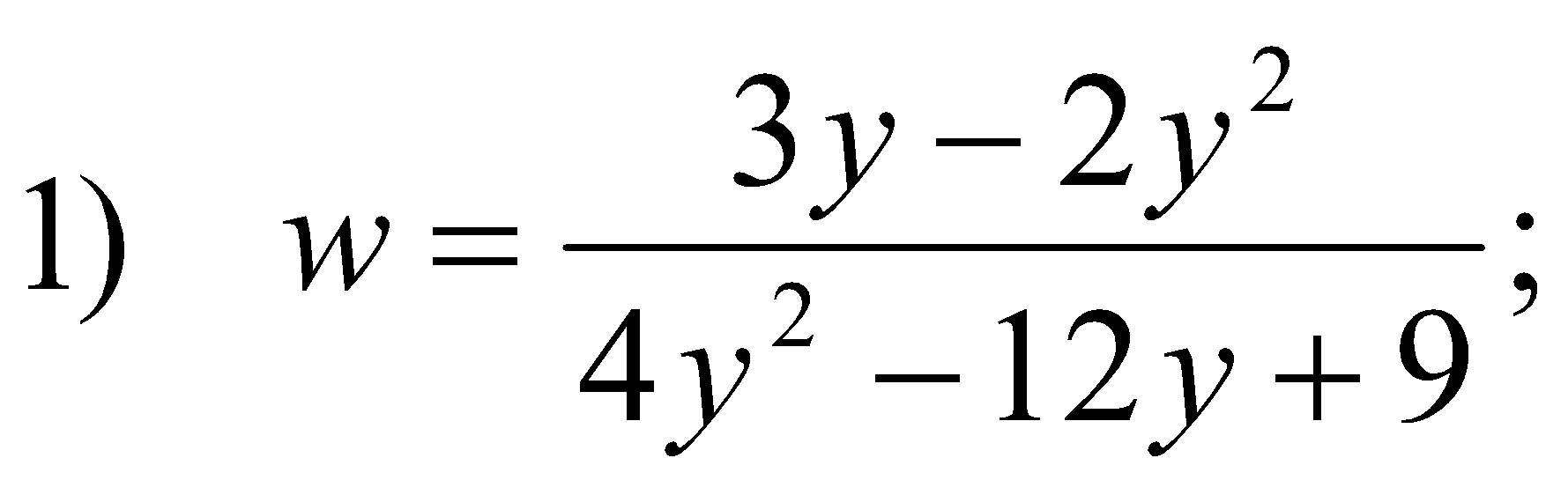




3) Составить программу перевода радианной меры угла в градусы, минуты и секунды.

4) Дано трехзначное число. Вывести вначале его последнюю цифру (единицы), а затем — его среднюю цифру (десятки).

5) Даны координаты поля шахматной доски *x*, *y* (целые числа, лежащие в диапазоне 1–8). Учитывая, что левое нижнее поле доски (1, 1) является черным, проверить истинность высказывания: «Данное поле является белым».

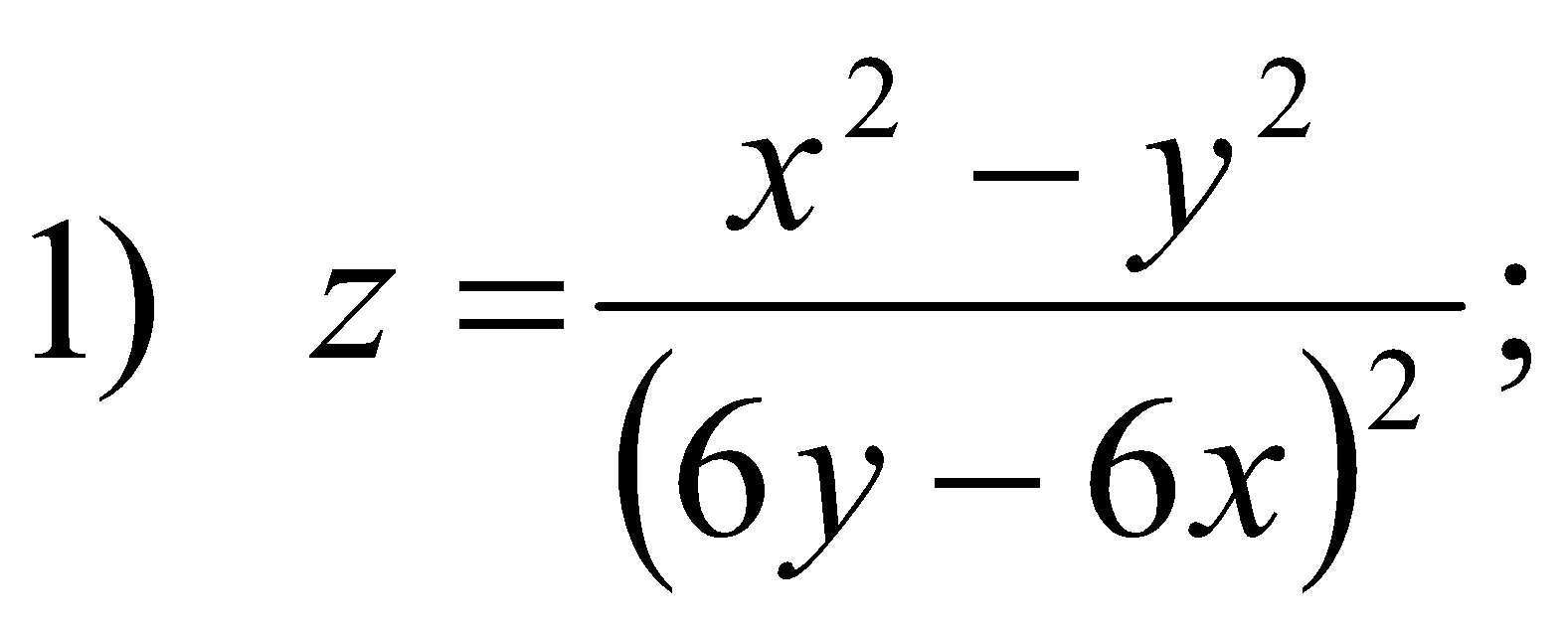
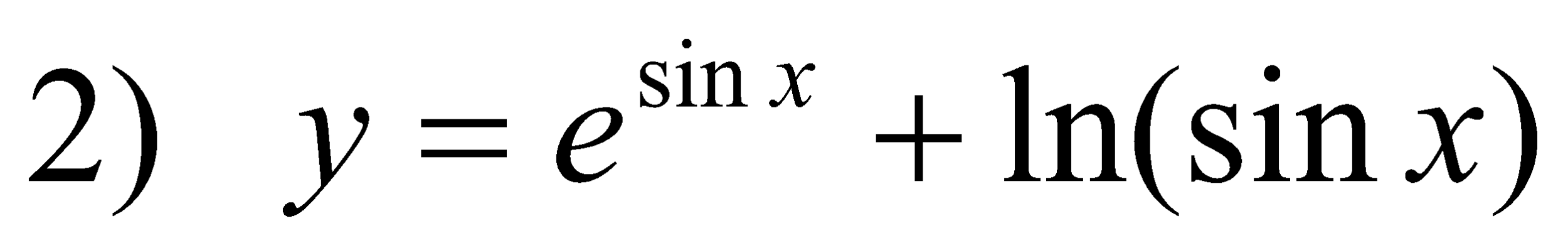
**Вариант 8**

3) У американского писателя-фантаста Рэя Бредбери есть роман "450 градусов по Фаренгейту". Какой температуре по шкале Цельсия соответствует указанное в названии значение?

4) Дано трехзначное число. Найти сумму и произведение его цифр.

5) Даны целые числа *a*, *b*, *c*. Проверить истинность высказывания: «Существует треугольник со сторонами *a*, *b*, *c*».

**Вариант 9**

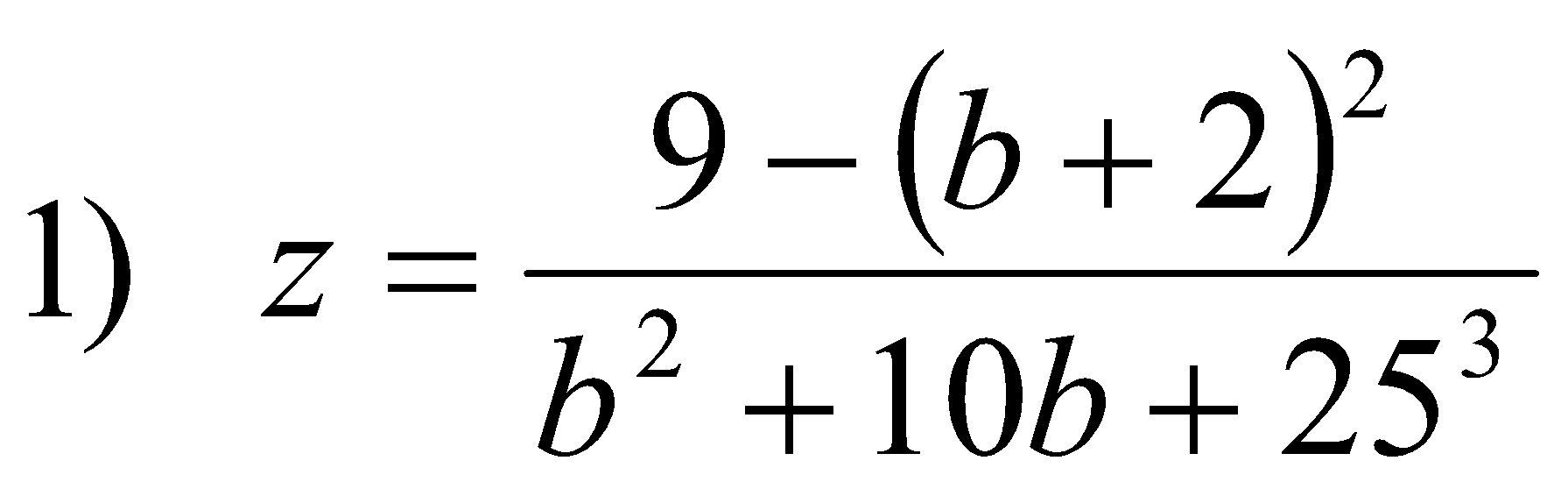
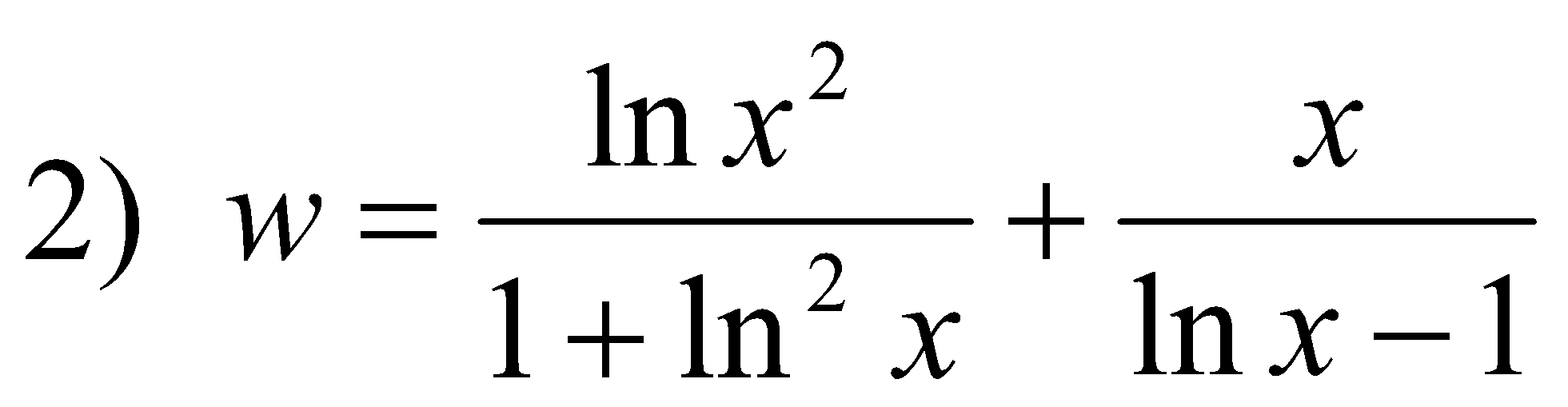


3)Дана длина ребра куба. Найти площадь грани, площадь полной поверхности и объем этого куба.

4) Дано трехзначное число. Вывести число, полученное при прочтении исходного числа справа налево.

5) Даны целые числа *a*, *b*, *c*, являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами *a*, *b*, *c* является прямоугольным».

**Вариант 10**

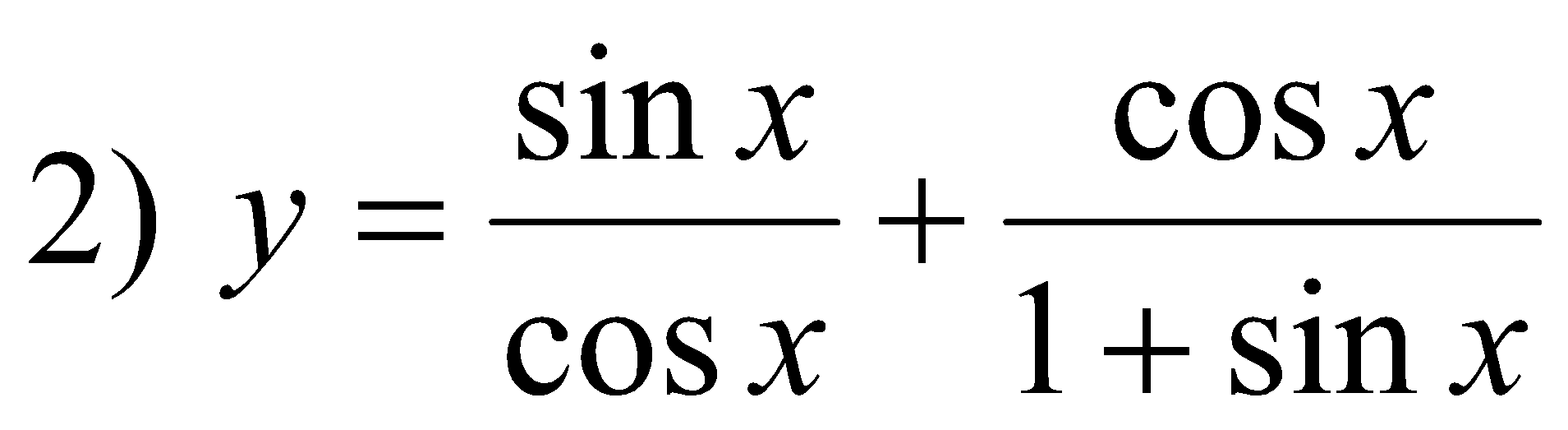
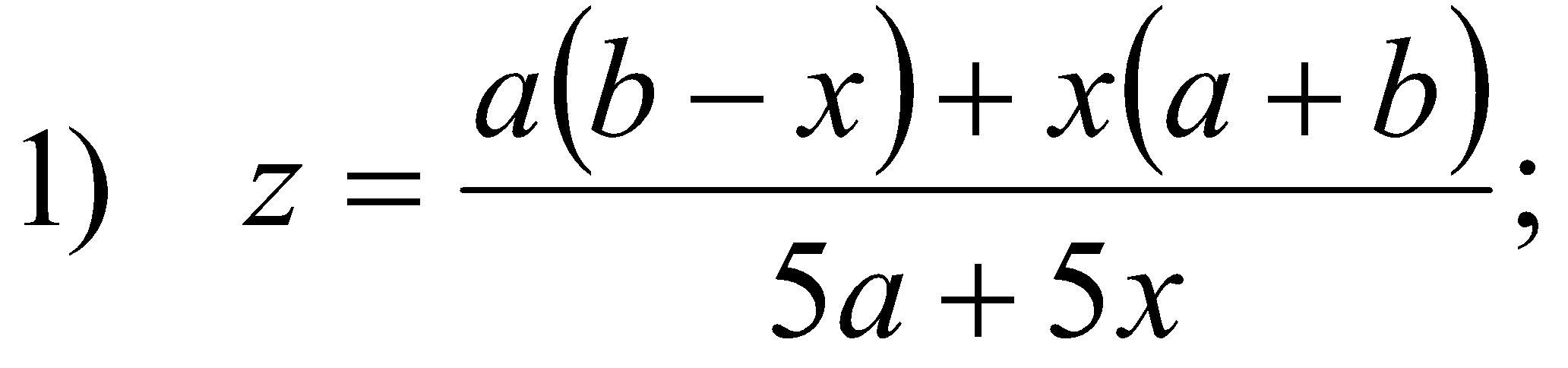


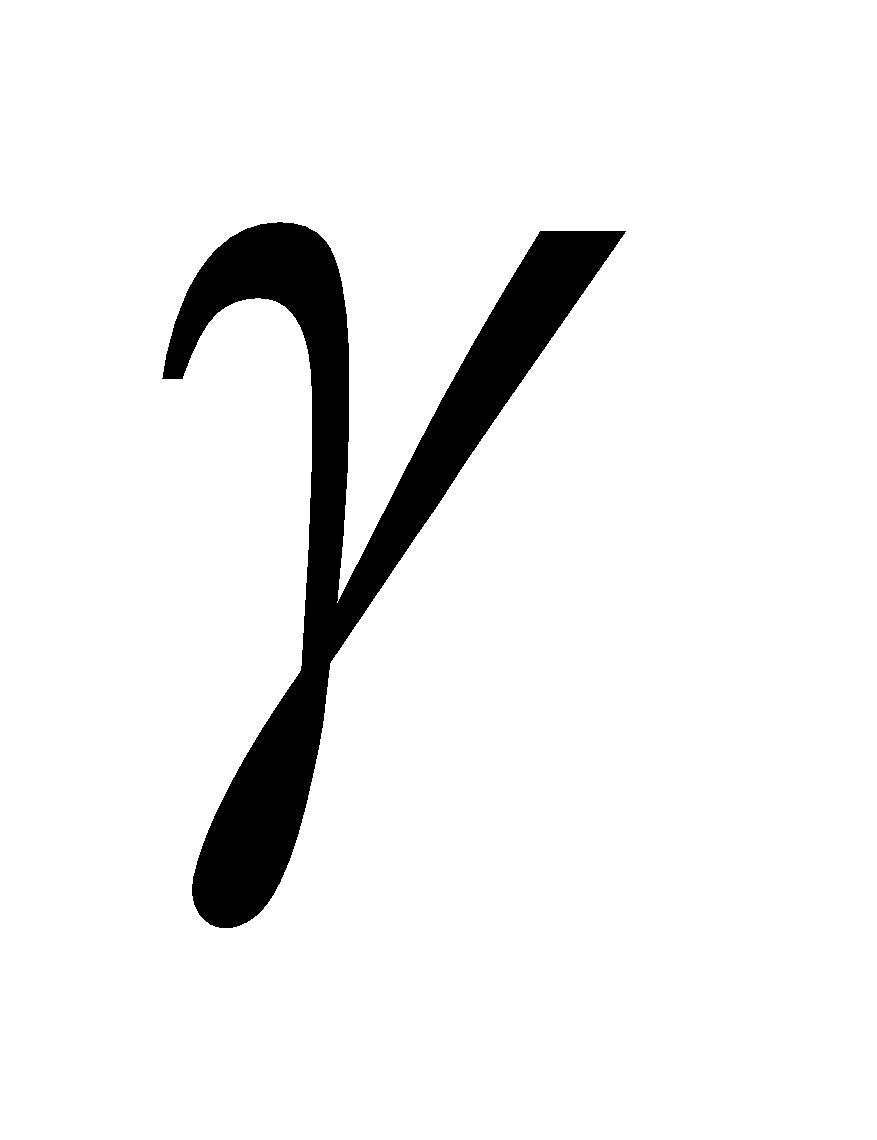
3) Три сопротивления соединены параллельно. Найдите сопротивление соединения.

4) Дано трехзначное число. В нем зачеркнули первую слева цифру и приписали ее справа. Вывести полученное число.

5) Дано целое число *A*. Проверить истинность высказывания: «Число *A* является положительным».

**Вариант 11**

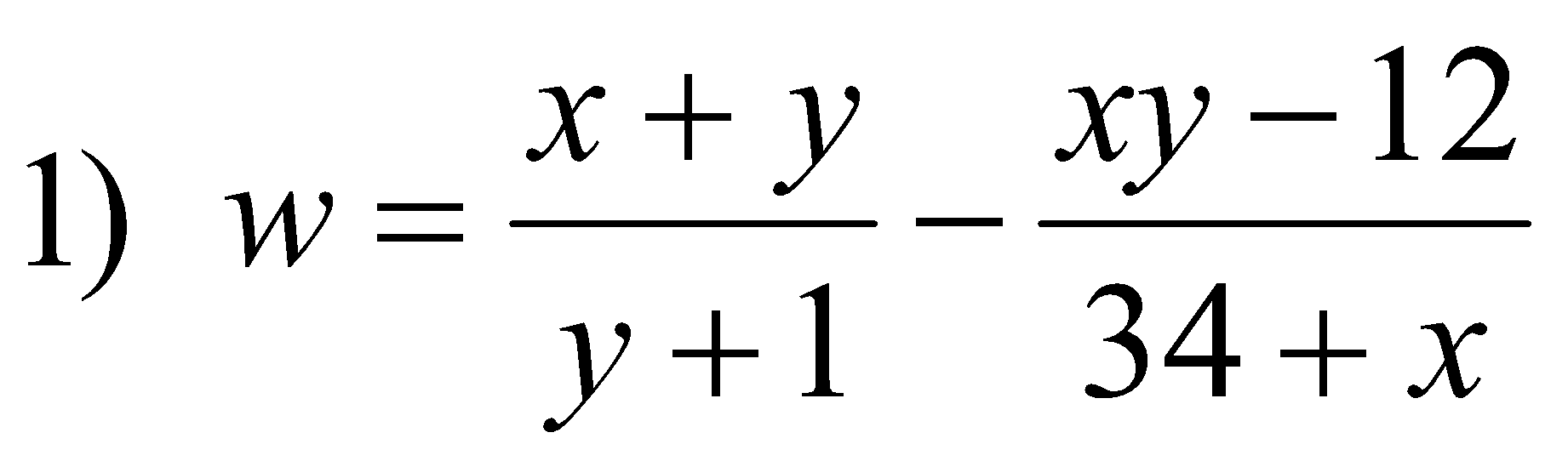


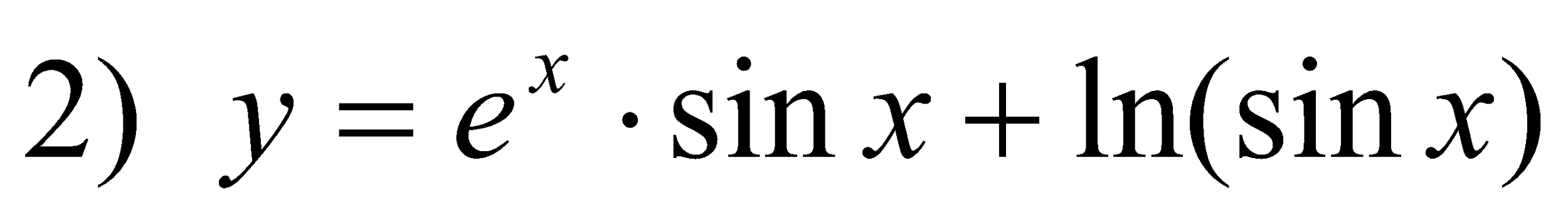
3) Найти площадь треугольника, две стороны которого равны a и b, а угол между этими сторонами равен 

4) Дано трехзначное число. В нем зачеркнули первую справа цифру и приписали ее слева. Вывести полученное число.

5) Дано целое число *A*. Проверить истинность высказывания: «Число *A* является нечетным».

**Вариант 12**



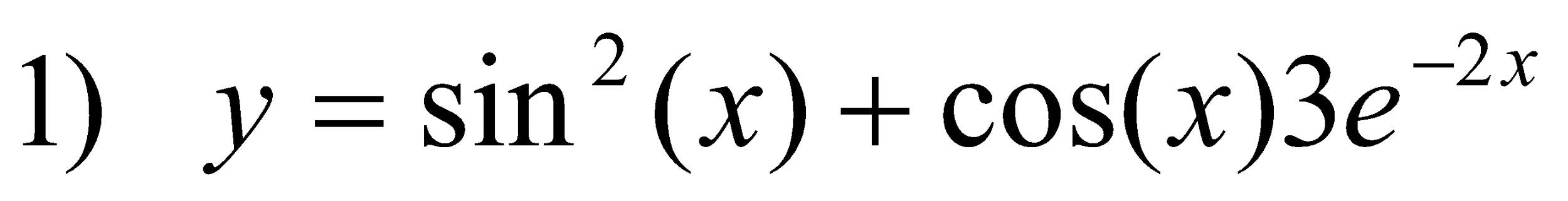
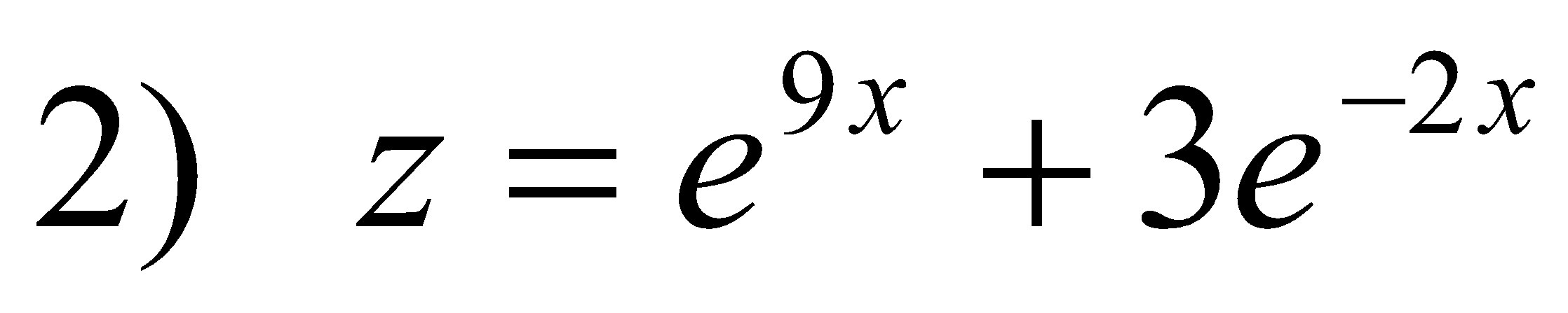


3) Дано a. Не используя никаких функций и никаких операций, кроме умножения, получить a8 за три операции и a10 за четыре операции.

4) Дано трехзначное число. Вывести число, полученное при перестановке цифр сотен и десятков исходного числа (например, 123 перейдет в 213).

5) Дано целое число *A*. Проверить истинность высказывания: «Число *A* является четным».

**Вариант 13**

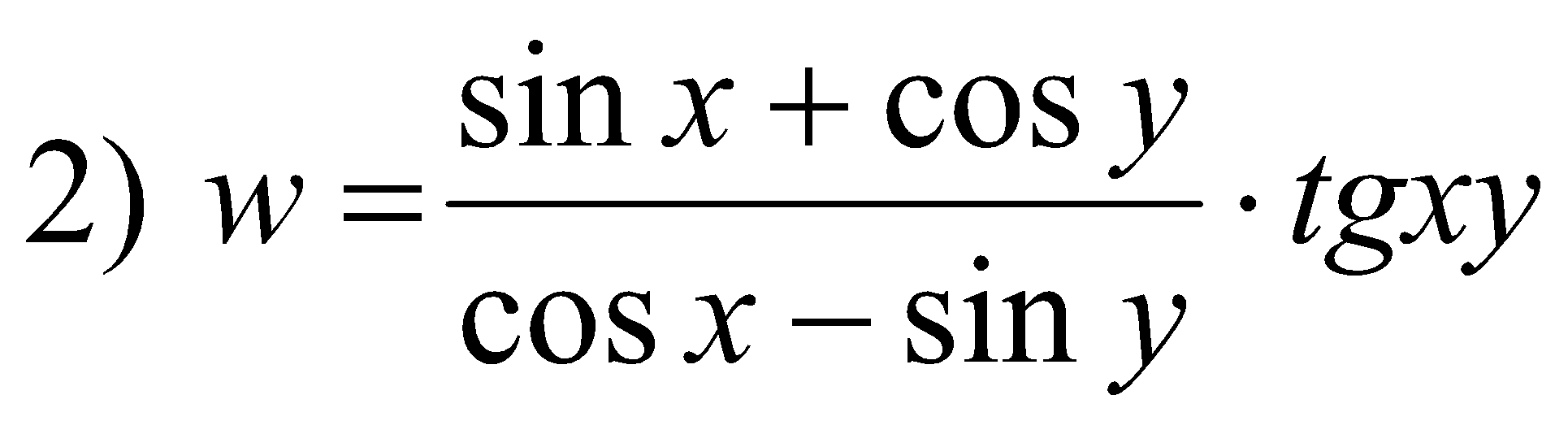
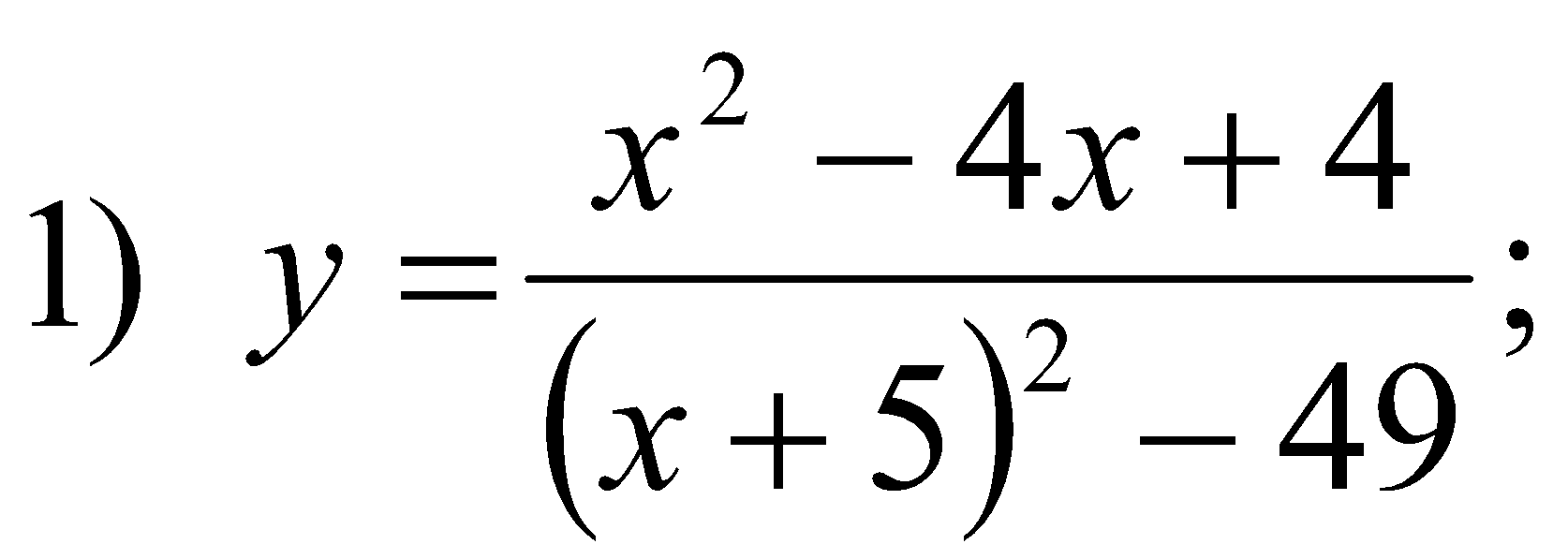


3)Вычислить расстояние между двумя точками с данными координатами x1, y1 и x2, y2.

4) Дано трехзначное число. Вывести число, полученное при перестановке цифр десятков и единиц исходного числа (например, 123 перейдет в 132).

5) Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Справедливы неравенства *A* > 2 и *B*<=3».

**Вариант 14**

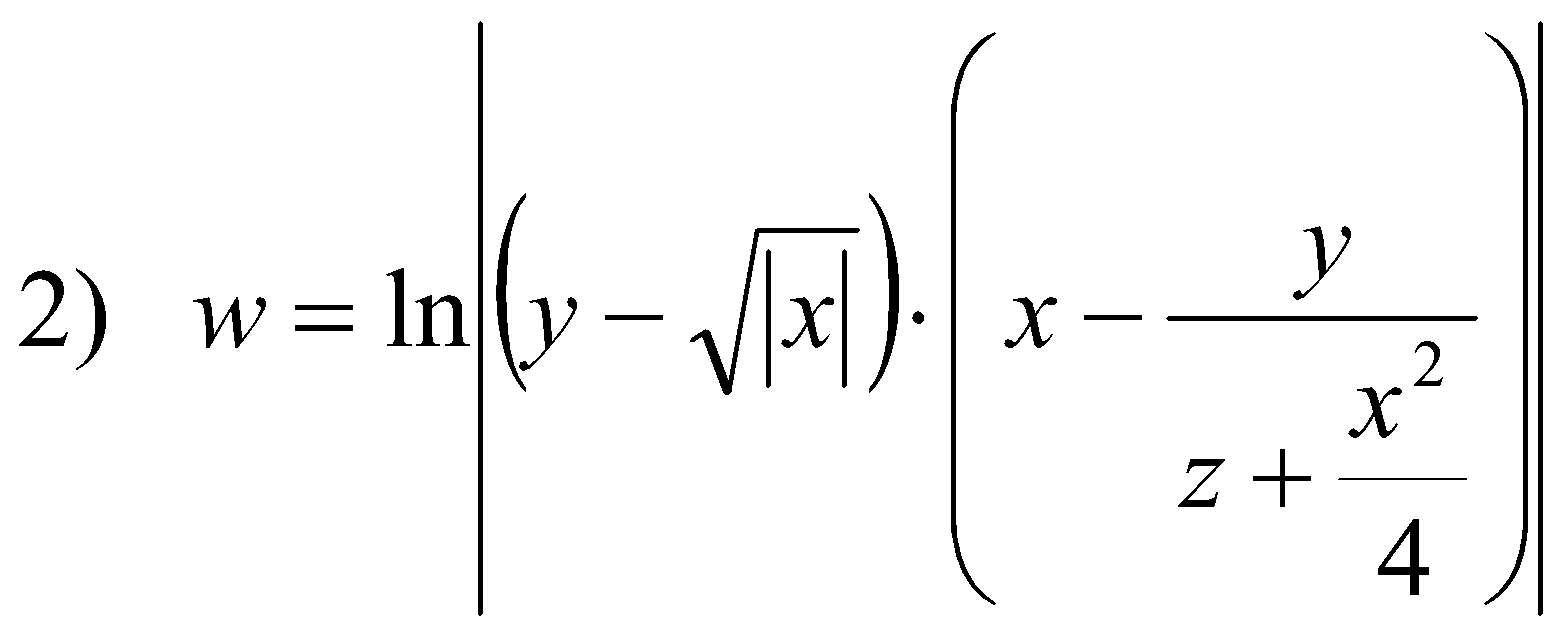


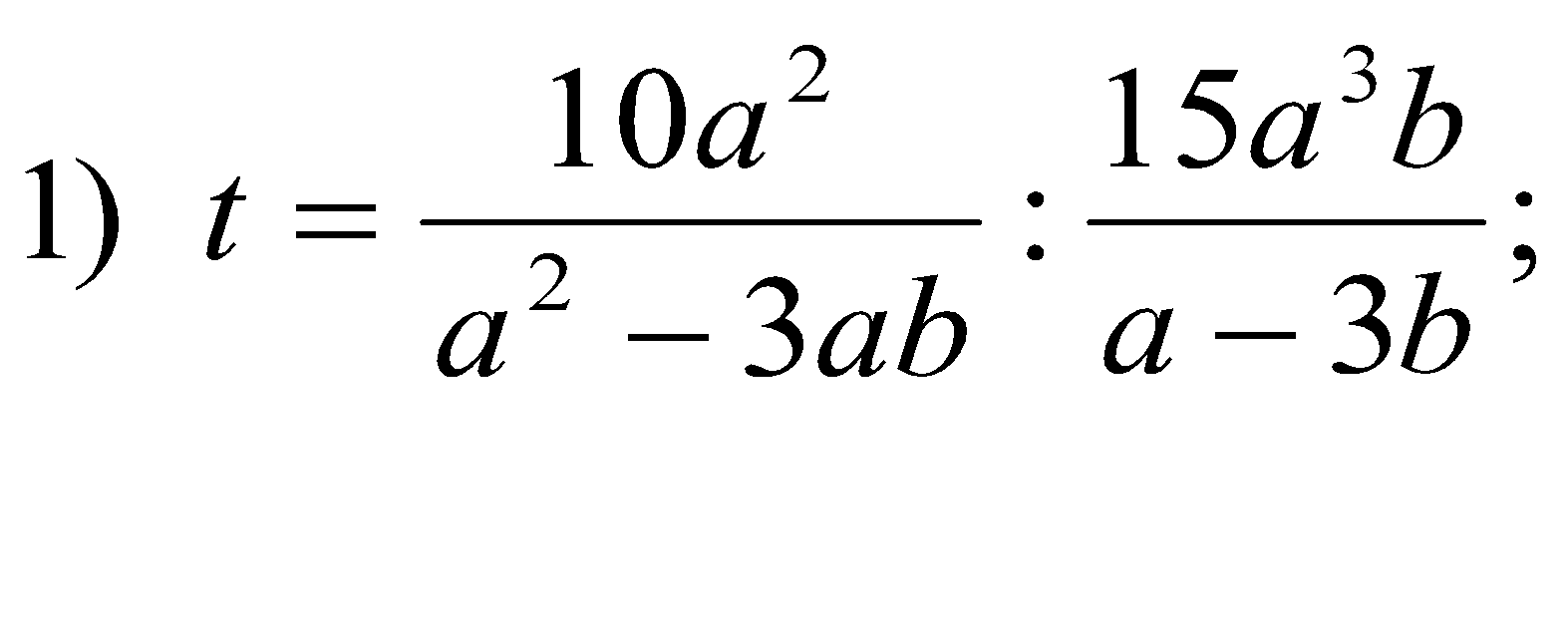
3) Вычислить периметр и площадь прямоугольного треугольника по данным длинам двух катетов а и в.

4) Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду сотен в записи этого числа.

5) Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Справедливы неравенства *A*<=0 или *B* < –2».

**Вариант 15**



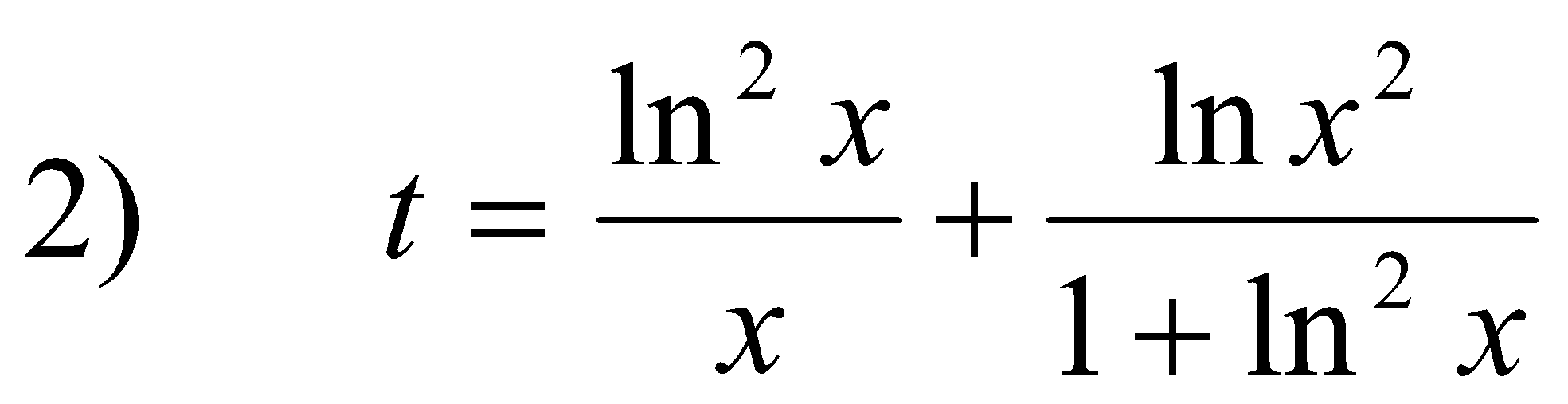
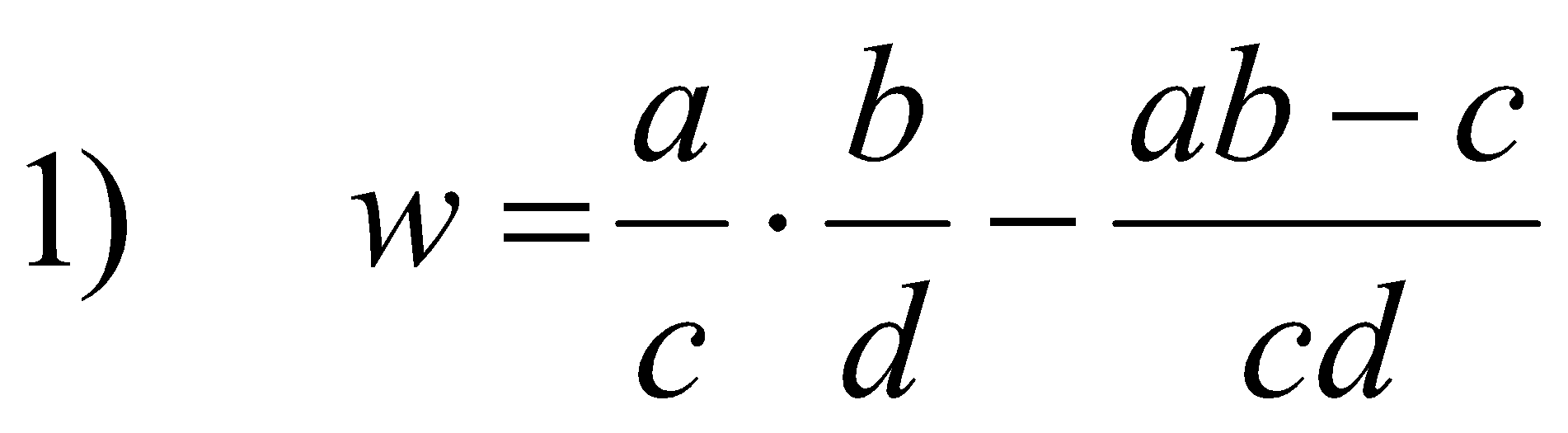


3)Треугольник задан величинами углов и радиусом описанной окружности. Найти стороны треугольника.

4) Дано целое число, большее 999. Используя одну операцию деления нацело и одну операцию взятия остатка от деления, найти цифру, соответствующую разряду тысяч в записи этого числа.

5) Даны три целых числа: *A*, *B*, *C*. Проверить истинность высказывания: «Справедливо двойное неравенство *A* < *B* < *C*».

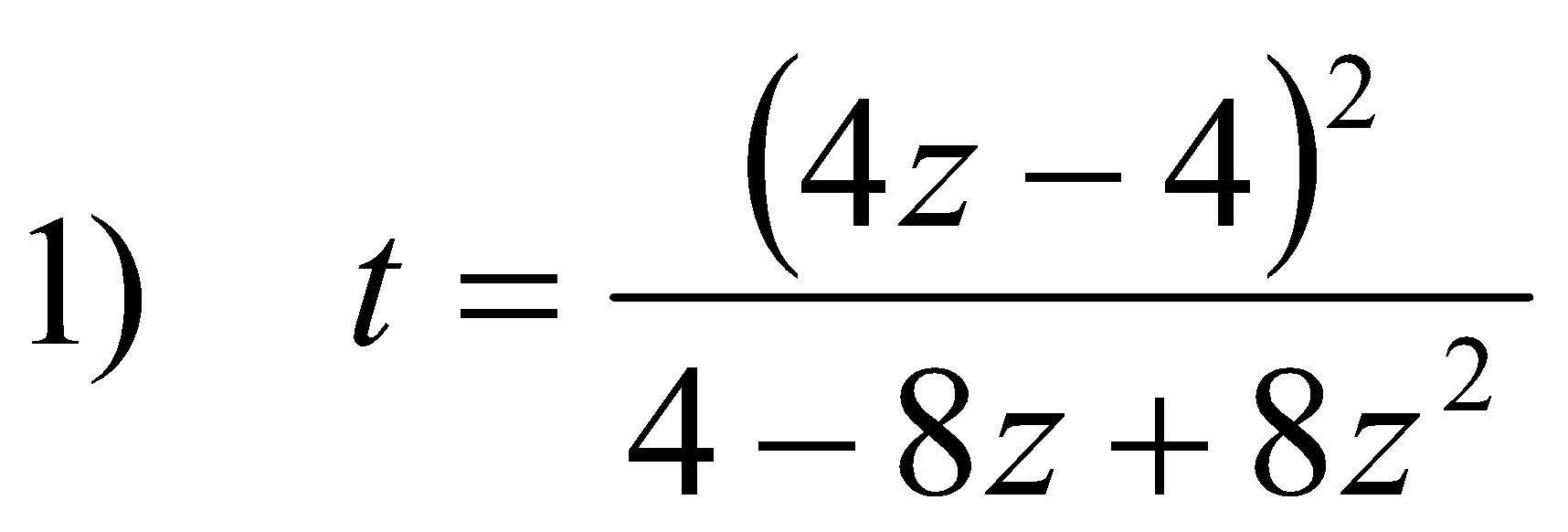
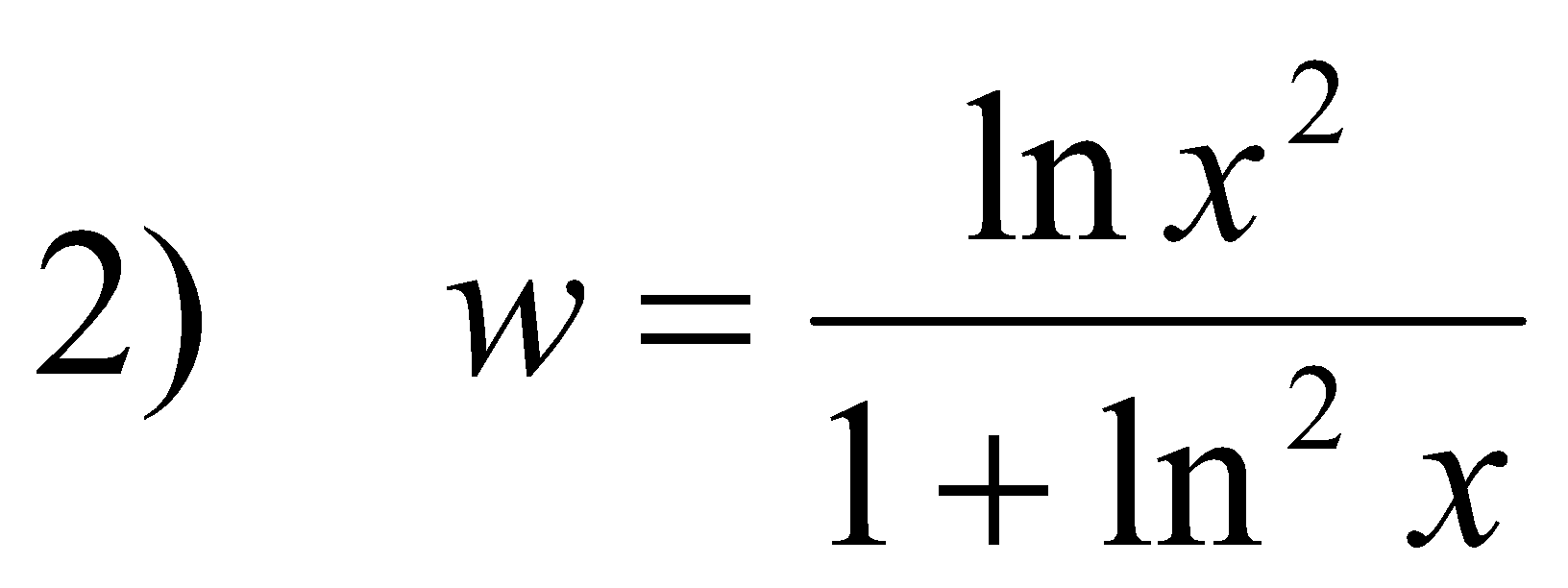
**Вариант 16**



3)Найти площадь кольца, внутренний радиус которого равен r, а внешний - заданному числу R (R>r).

4) С начала суток прошло *N* секунд (*N* — целое). Найти количество полных минут, прошедших с начала суток.

5) Даны три целых числа: *A*, *B*, *C*. Проверить истинность высказывания: «Число *B* находится между числами *A* и *C*».

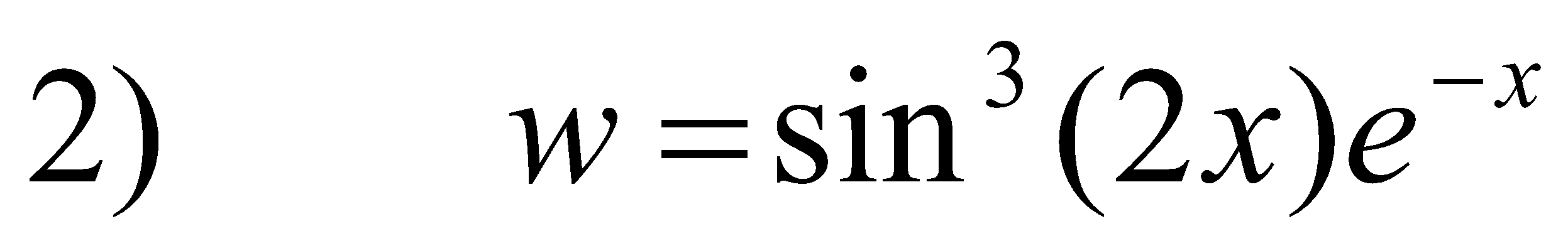
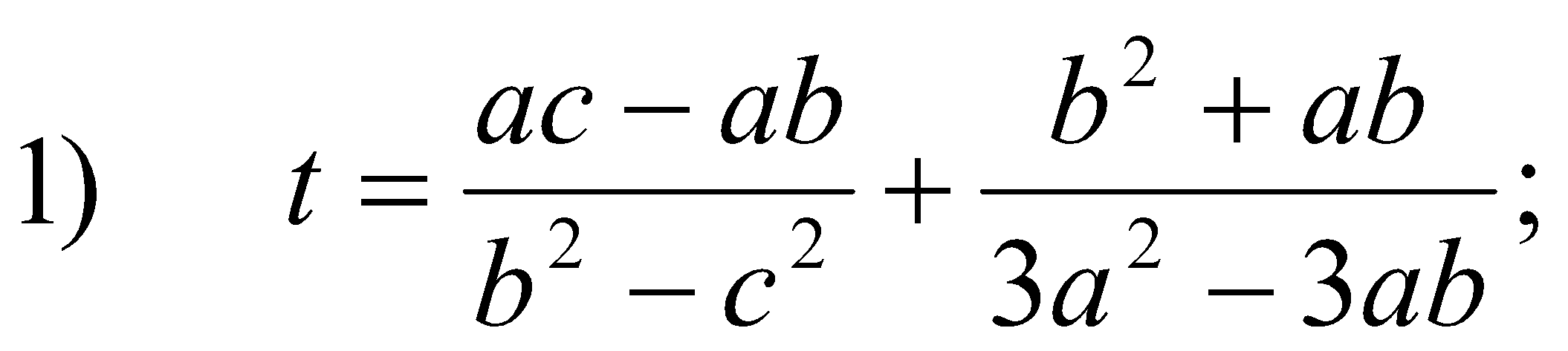
**Вариант 17** 

3)Дано x. Получить значения -2x+3x2-4x3 и 1+2x+3x2+4x3. Позаботиться об экономии операций.

4) С начала суток прошло *N* секунд (*N* — целое). Найти количество полных часов, прошедших с начала суток.

5) Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Каждое из чисел *A* и *B* нечетное».

**Вариант 18**

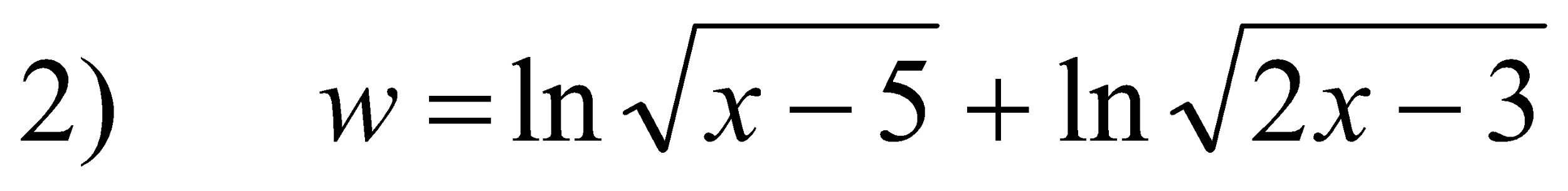
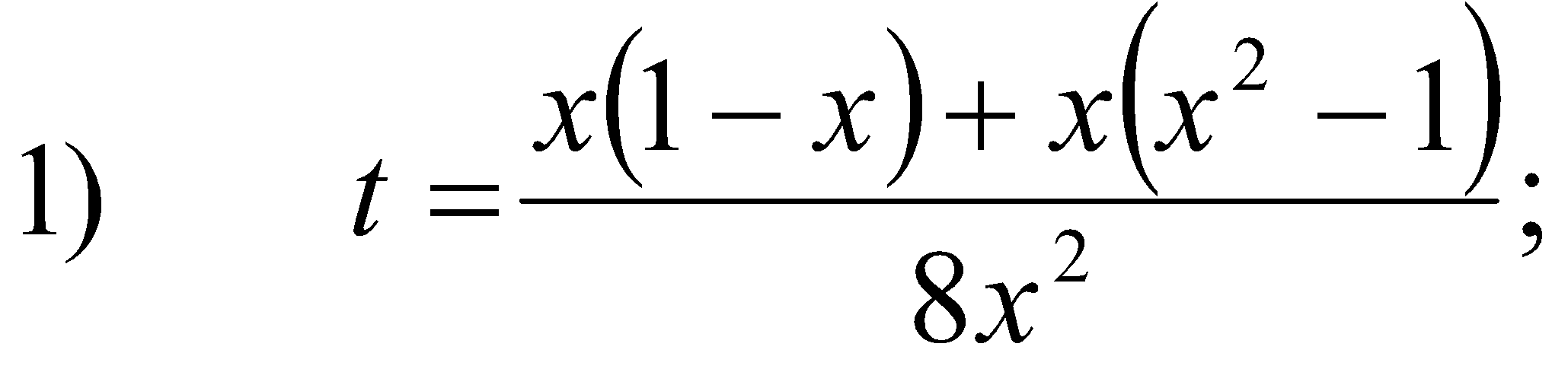


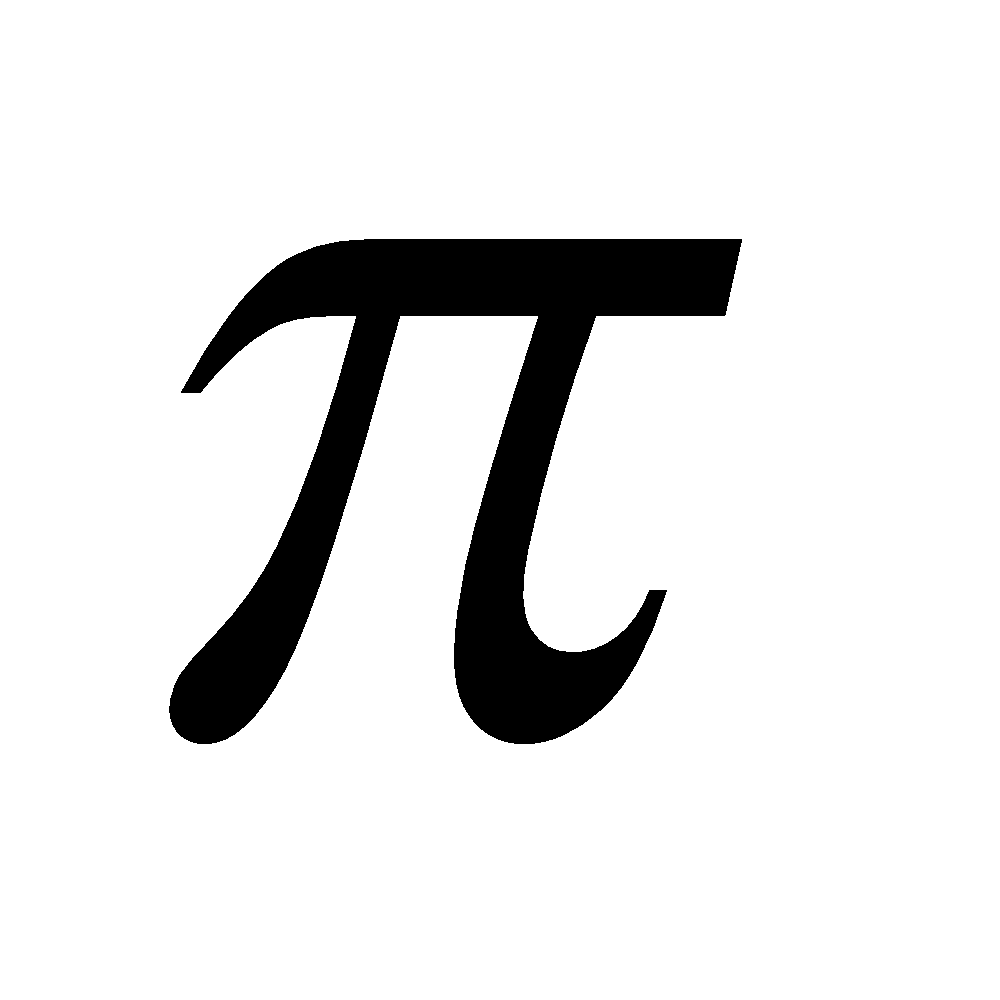
3) Полторы кошки за полтора часа съедают полторы мышки. Сколько мышек съедят X кошек за Y часов.

4) С начала суток прошло *N* секунд (*N* — целое). Найти количество секунд, прошедших с начала последней минуты.

5) Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Хотя бы одно из чисел *A* и *B* нечетное».

**Вариант 19**

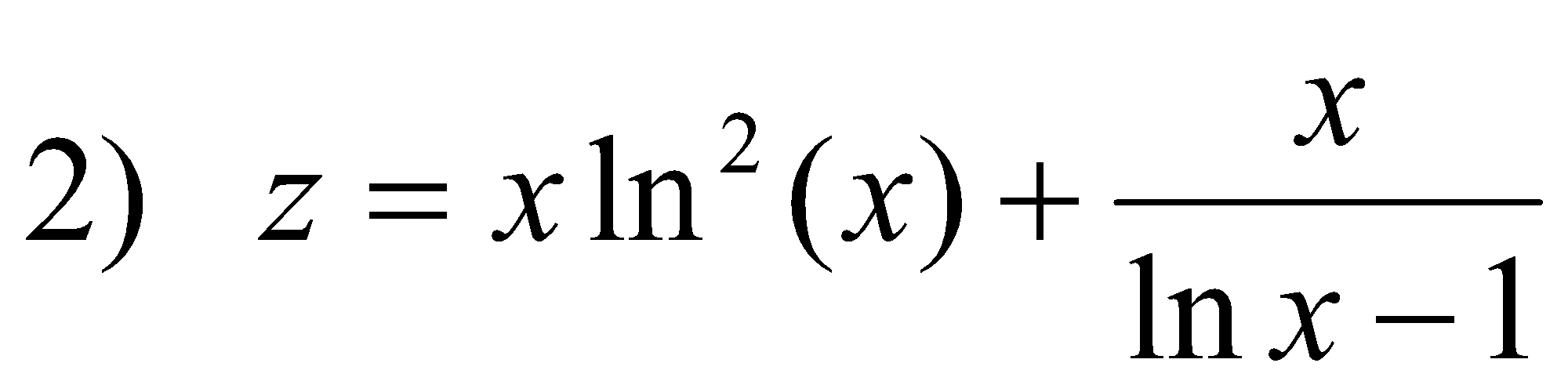
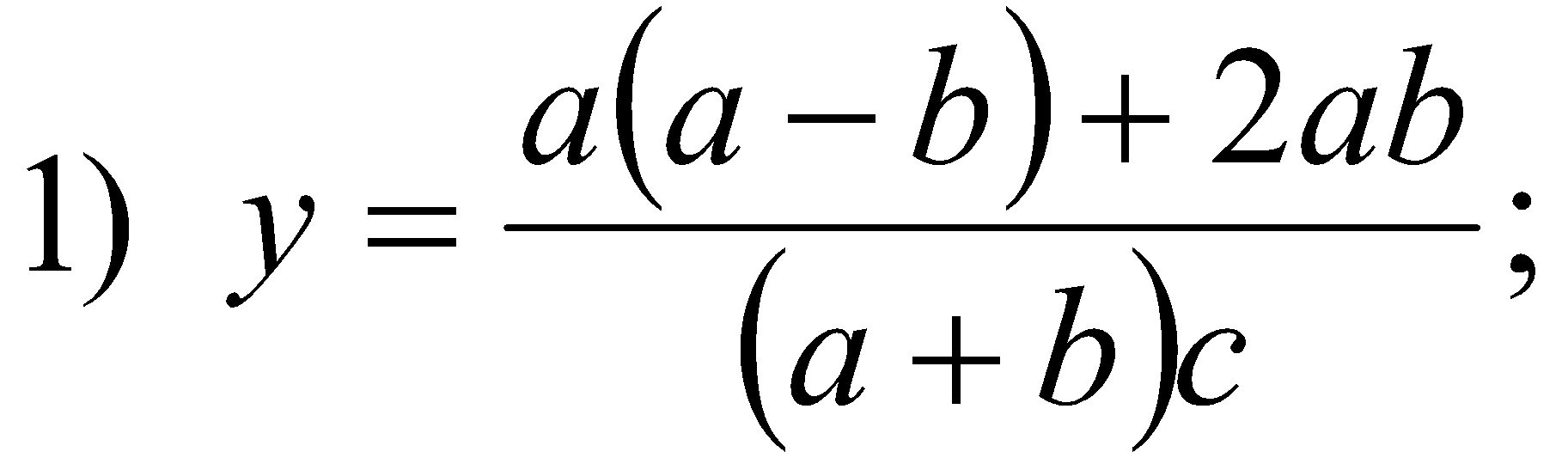


3) Написать программу, которая выводит на экран первые четыре степени числа . 

4) С начала суток прошло *N* секунд (*N* — целое). Найти количество секунд, прошедших с начала последнего часа.

5) Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Ровно одно из чисел *A* и *B* нечетное».

**Вариант 20**

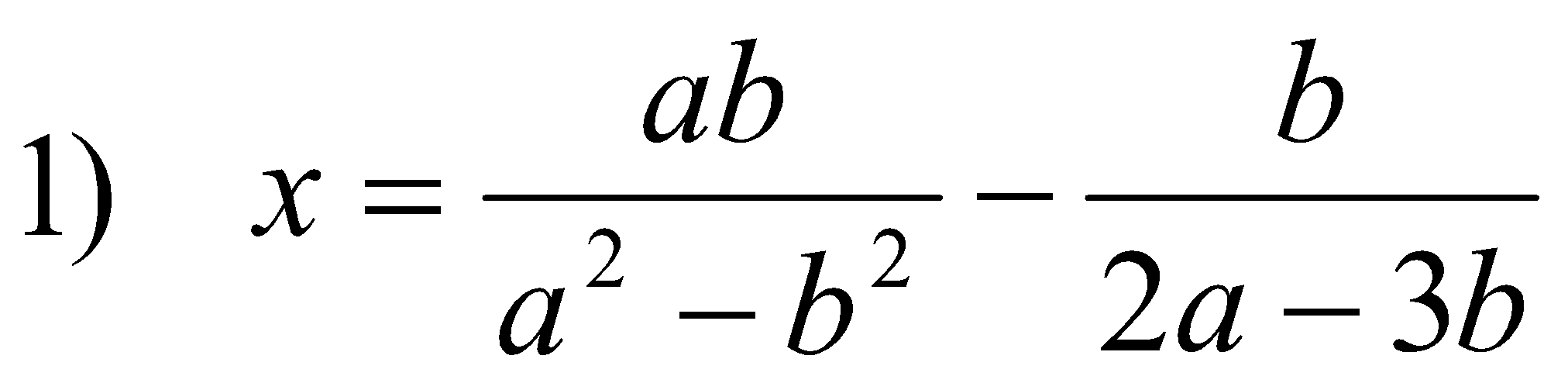
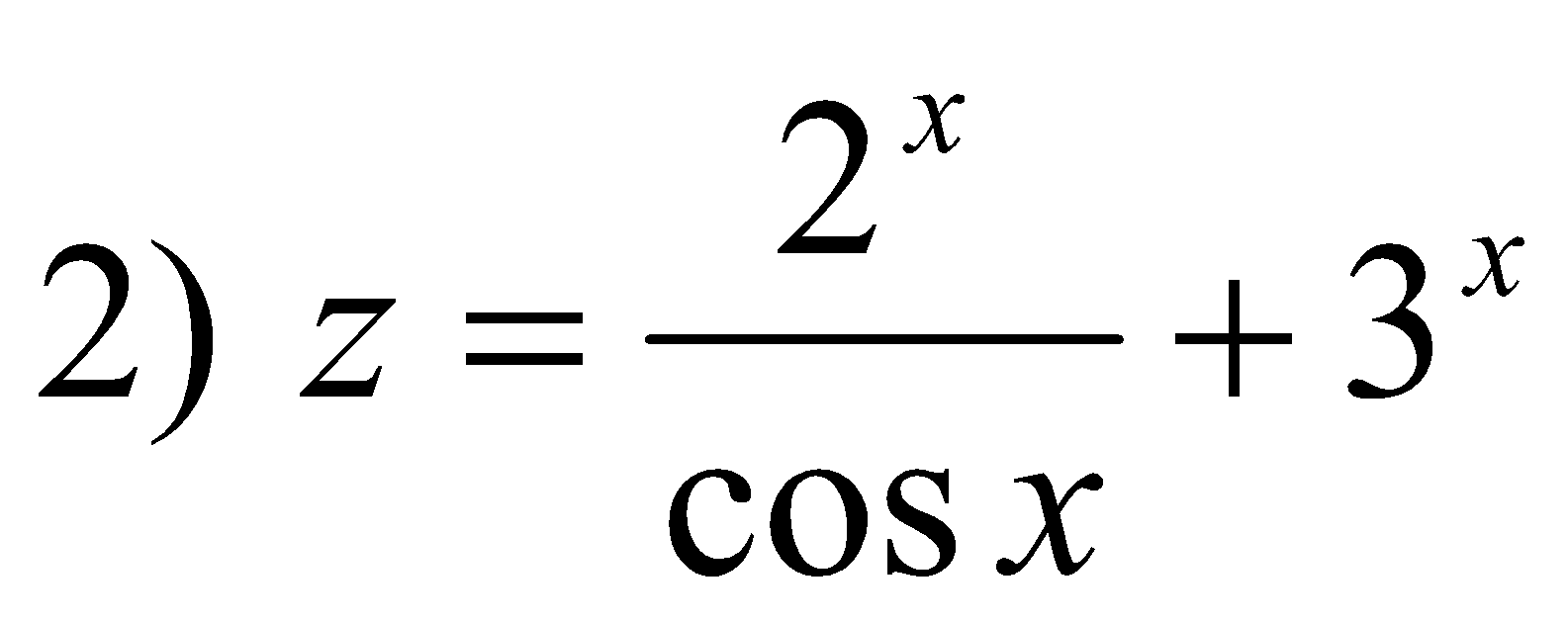


3) Вычислить длину окружности и площадь круга одного и того же заданного радиуса R.

4) С начала суток прошло *N* секунд (*N* — целое). Найти количество полных минут, прошедших с начала последнего часа.

5) Даны два целых числа: *A*, *B*. Проверить истинность высказывания: «Числа *A* и *B* имеют одинаковую четность».

**Вариант 21**

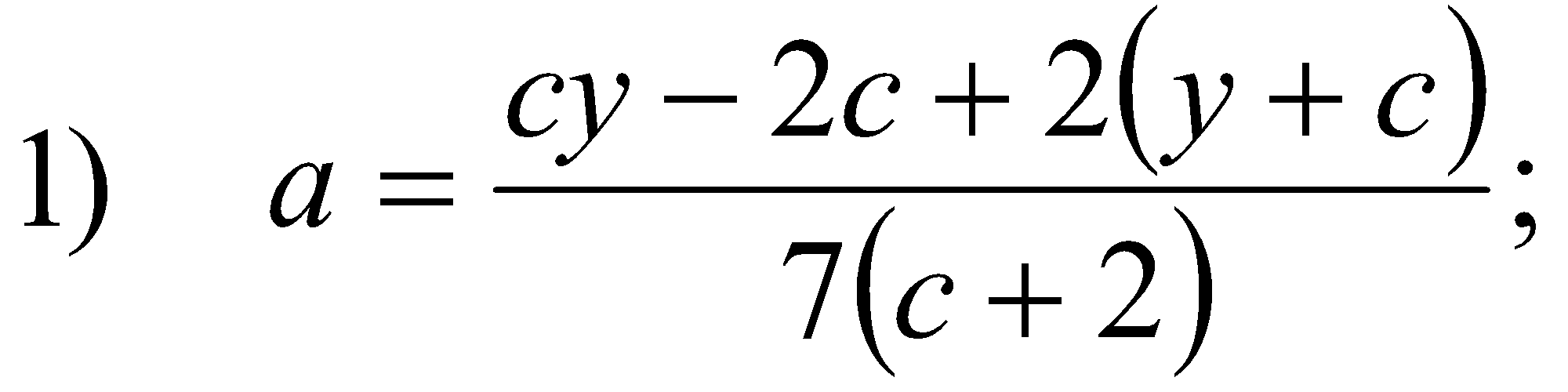


3)Составить программу вычисления объема цилиндра и конуса, которые имеют одинаковую высоту H и одинаковый радиус основания R.

4) Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, …, 6 — суббота. Дано целое число *K*, лежащее в диапазоне 1–365. Определить номер дня недели для *K*-го дня года, если известно, что в этом году 1 января было понедельником.

5) Даны три целых числа: *A*, *B*, *C*. Проверить истинность высказывания: «Каждое из чисел *A*, *B*, *C* положительное».

**Вариант 22**



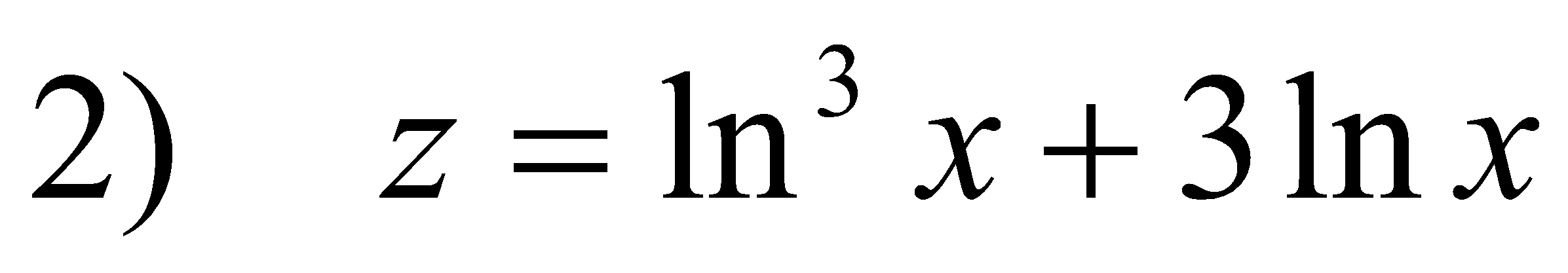
3)Даны два действительных числа x и y. Вычислить их сумму, разность, произведение и частное.

4) Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, …, 6 — суббота. Дано целое число *K*, лежащее в диапазоне 1–365. Определить номер дня недели для *K*-го дня года, если известно, что в этом году 1 января было четвергом.

5) Даны три целых числа: *A*, *B*, *C*. Проверить истинность высказывания: «Хотя бы одно из чисел *A*, *B*, *C* положительное».

**Вариант 23**



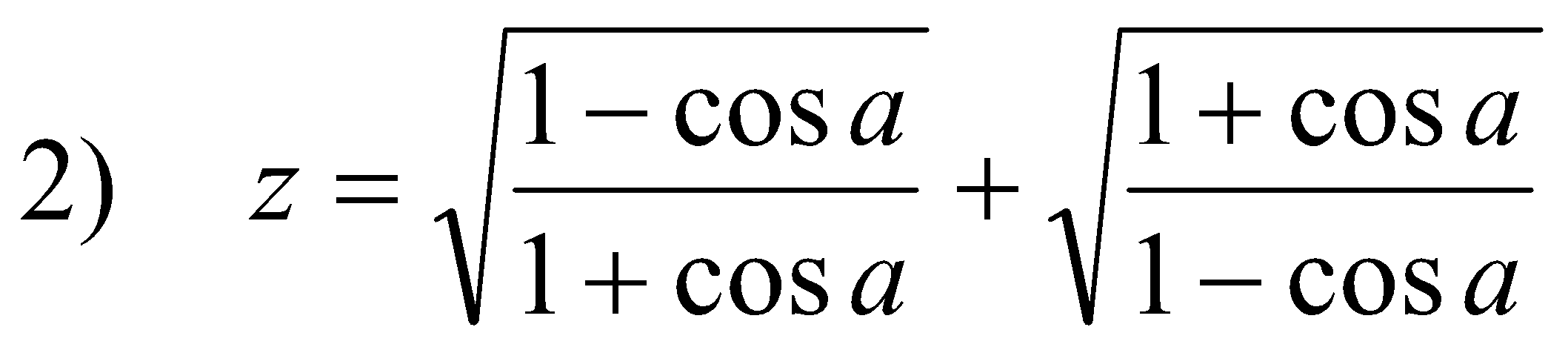
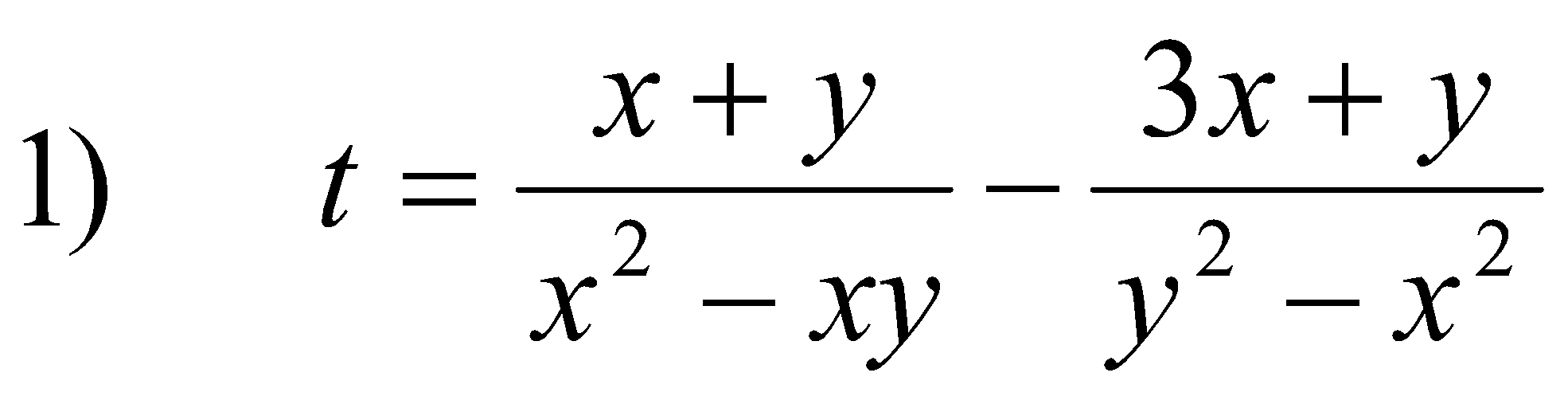


3)Вычислить корни квадратного уравнения ax2+bx+c=0, заданного коэффициентами a,b и c (предполагается, что a не равно нулю и что дискриминант уравнения неотрицателен).

4) Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, …, 6 — суббота, 7 — воскресенье. Дано целое число *K*, лежащее в диапазоне 1–365. Определить номер дня недели для *K*-го дня года, если известно, что в этом году 1 января было вторником.

5) Даны три целых числа: *A*, *B*, *C*. Проверить истинность высказывания: «Ровно одно из чисел *A*, *B*, *C* положительное».

**Вариант 24**

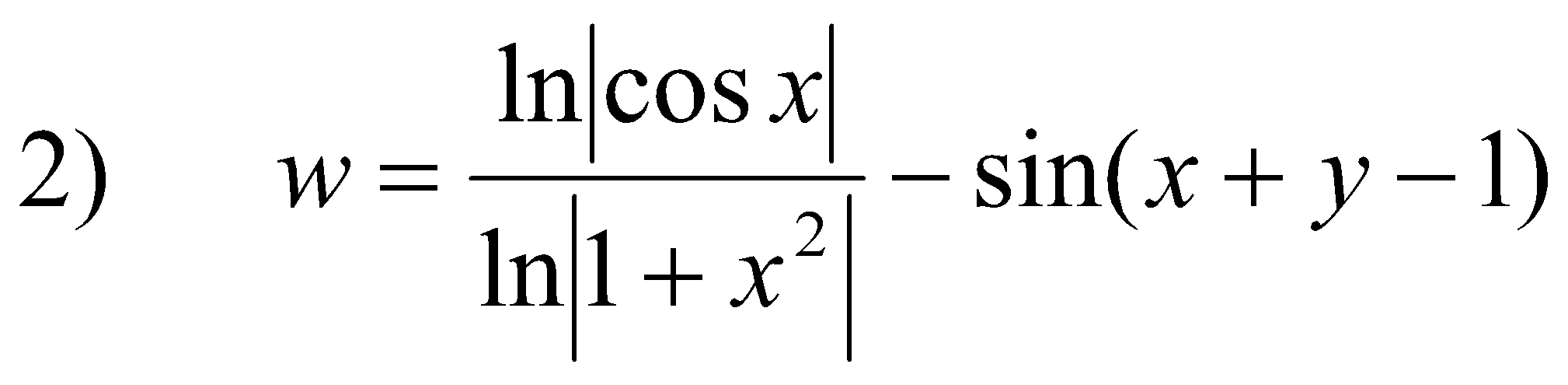
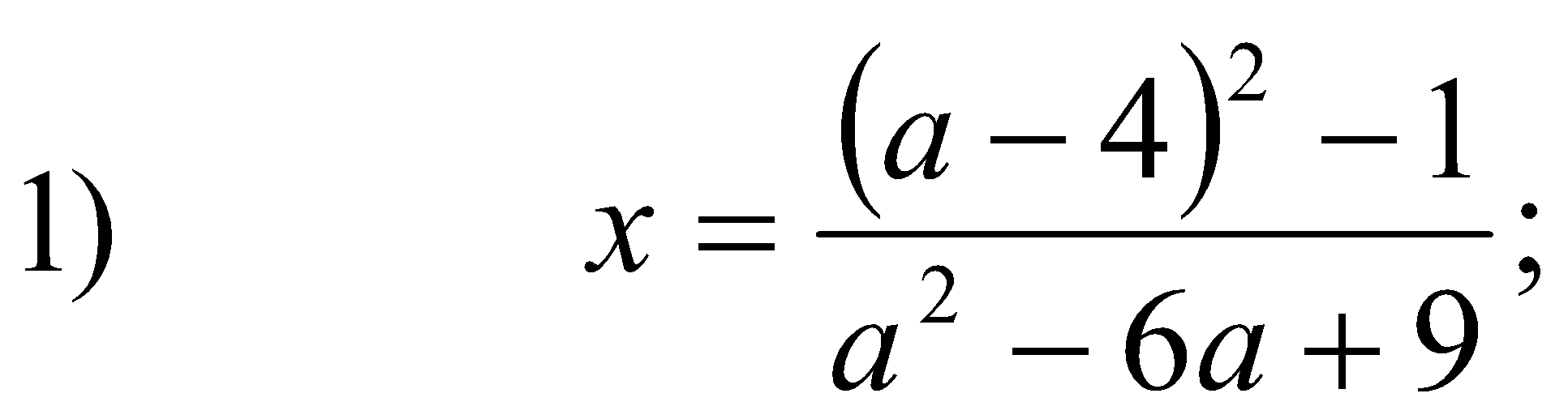


3)Дано действительное число x. Не пользуясь никакими другими арифметическими операциями, кроме умножения, сложения и вычитания, вычислить за минимальное число операций 2x4-3x3+4x2-5x+6.

4) Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, …, 6 — суббота, 7 — воскресенье. Дано целое число *K*, лежащее в диапазоне 1–365. Определить номер дня недели для *K*-го дня года, если известно, что в этом году 1 января было субботой.

5) Даны три целых числа: *A*, *B*, *C*. Проверить истинность высказывания: «Ровно два из чисел *A*, *B*, *C* являются положительными».

**Вариант 25**



3) Найти (в градусах) все углы треугольника со сторонами a, b, c.

4) Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, …, 6 — суббота, 7 — воскресенье. Дано целое число *K*, лежащее в диапазоне 1–365, и целое число *N*, лежащее в диапазоне 1–7. Определить номер дня недели для *K*-го дня года, если известно, что в этом году 1 января было днем недели с номером *N*.

5) Дано целое положительное число. Проверить истинность высказывания: «Данное число является четным двузначным».

**Вариант 26**

1) *y =*  2) *y =*

3) Даны стороны прямоугольника *a* и *b*. Найти его площадь *S* = *a*·*b* и периметр *P* = 2·(*a* + *b*).

4) Даны целые положительные числа *A*, *B*, *C*. На прямоугольнике размера *A* х *B* размещено максимально возможное количество квадратов со стороной *C* (без наложений). Найти количество квадратов, размещенных на прямоугольнике, а также площадь незанятой части прямоугольника.

5) Дано целое положительное число. Проверить истинность высказывания: «Данное число является нечетным трехзначным».

**Вариант 27**

1) *y =*  2) *y =*

3) Дан диаметр окружности *d*. Найти ее длину *L* = ·*d*. В качестве значения π использовать 3.14.

4) Дан номер некоторого года (целое положительное число). Определить соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.

5) Проверить истинность высказывания: «Среди трех данных целых чисел есть хотя бы одна пара совпадающих».

**Вариант 28**

1) *y =*  2) *y =*

3) Дана длина ребра куба *a*. Найти объем куба *V* = *a*3 и площадь его поверхности *S* = 6·*a*2.

4) В трехзначном числе *x* зачеркнули его вторую цифру. Когда к образованному при этом двузначному числу слева приписали вторую цифру числа *x*, то получилось число 546. Найти число *x*.

5) Проверить истинность высказывания: «Среди трех данных целых чисел есть хотя бы одна пара взаимно противоположных».

**Вариант 29**

1) *y =*  2) *y =*

3) Даны длины ребер *a*, *b*, *c* прямоугольного параллелепипеда. Найти его объем *V* = *a·b·c* и площадь поверхности *S* = 2·(*a·b* + *b·c* + *a·c*).

4) В трехзначном числе *x* зачеркнули первую цифру. Когда полученное число  
умножили на 10, а произведение сложили с первой цифрой числа *x*, то получилось число *n*. По заданному *n* найти число *x* (значение *n* вводится с клавиатуры, 1 *≤ n ≤* 999).

5) Дано трехзначное число. Проверить истинность высказывания: «Все цифры данного числа различны».

**Вариант 30**

1) *y =*  2) *y =*

3) Найти длину окружности *L* и площадь круга *S* заданного радиуса *R*:

*L* = 2··*R*,        *S* = ·*R*2.

В качестве значения π использовать 3.14.

4) В трехзначном числе *x* зачеркнули его вторую цифру. Когда к образованному  
при этом двузначному числу слева приписали вторую цифру числа *x*, то получилось число *n*. По заданному *n* найти число *n* (значение *n* вводится с клавиатуры, 10 *≤ n ≤* 999 и при этом число десятков в *n* не равно нулю).

5) Дано трехзначное число. Проверить истинность высказывания: «Цифры данного числа образуют возрастающую последовательность».

**Вариант 31**

1) *y =*  2) *y =*

3) Даны два ненулевых числа. Найти сумму, разность, произведение и частное их квадратов.

4) В трехзначном числе *x* зачеркнули его вторую цифру. Когда к образованному при этом двузначному числу справа приписали вторую цифру числа *x*, то получилось число 456. Найти число *x*.

5) Дано трехзначное число. Проверить истинность высказывания: «Цифры данного числа образуют возрастающую или убывающую последовательность».

**Вариант 32**

1) *y* = 2 *x*2 + 3 *x* − 1 2) *y = x - x*3

3) Даны два ненулевых числа. Найти сумму, разность, произведение и частное их модулей.

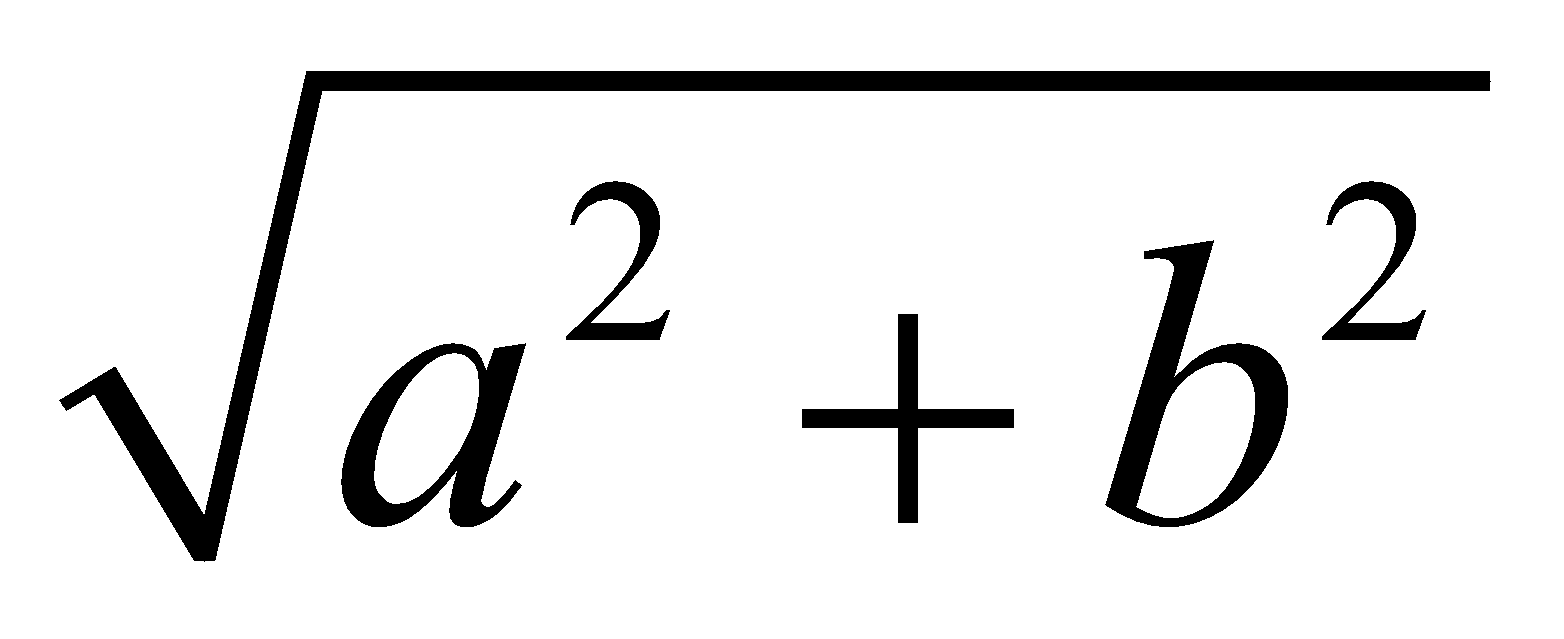
4) В трехзначном числе *x* зачеркнули его вторую цифру. Когда к образованному  
при этом двузначному числу справа приписали вторую цифру числа *x*, то получилось число *n*. По заданному *n* найти число *x* (значение *n* вводится с клавиатуры, 100 *≤ n ≤* 999).

5) Дано четырехзначное число. Проверить истинность высказывания: «Данное число читается одинаково слева направо и справа налево».

**Вариант 33**

1) *y* = arcsin *x+*2*x*2 2) *y* = 2*x*2+ 3

3) Даны катеты прямоугольного треугольника *a* и *b*. Найти его гипотенузу *c* и периметр *P*:

*c* = ,        *P* = *a* + *b* + *c*.

*4)* В трехзначном числе *x* зачеркнули его последнюю цифру. Когда в оставшемся двузначном числе переставили цифры, а затем приписали к ним слева последнюю цифру числа *x*, то получилось число 654. Найти число *x*.

5) Даны числа *A*, *B*, *C* (число *A* не равно 0). Рассмотрев *дискриминант* *D* = *B*2 – 4·*A*·*C*, проверить истинность высказывания: «Квадратное уравнение *A*·*x*2 + *B*·*x* + *C* = 0 имеет вещественные корни».

**Вариант 34**

1) *y* = 2*x* + ln | sin *x* + 2 cos *x* | 2) *y* = *x*·(sin ln *x* − cos ln *x*)

3) Даны два круга с общим центром и радиусами *R*1 и *R*2 (*R*1 > *R*2). Найти площади этих кругов *S*1 и *S*2, а также площадь *S*3 кольца, внешний радиус которого равен *R*1, а внутренний радиус равен *R*2:

*S*1 = π·(*R*1)2,        *S*2 = π·(*R*2)2,        *S*3 = *S*1 – *S*2.

В качестве значения π использовать 3.14.

4) В трехзначном числе *x* зачеркнули его последнюю цифру. Когда в оставшемся двузначном числе переставили цифры, а затем приписали к ним слева последнюю цифру числа *x*, то получилось число *n*. По заданному *n* найти число *x* (значение *n* вводится с клавиатуры, 1 *≤ n ≤* 999 и при этом число единиц в *n* не равно нулю).

5) Даны числа *x*, *y*. Проверить истинность высказывания: «Точка с координатами (*x*, *y*) лежит во второй координатной четверти».

**Вариант 35**

1) *y* = arctg (sh *x*) + (sh *x*)·ln ch *x* 2) *y* = *ex*·(cos 2*x* + 2 sin 2*x*)

2) Дана длина *L* окружности. Найти ее радиус *R* и площадь *S* круга, ограниченного этой окружностью, учитывая, что *L* = 2·π·*R*,    *S* = π·*R*2. В качестве значения π использовать 3.14.

4) Из трехзначного числа *x* вычли его последнюю цифру. Когда результат разделили на 10, а к частному слева приписали последнюю цифру числа *x*, то получилось число 237. Найти число *x*.

5) Даны числа *x*, *y*. Проверить истинность высказывания: «Точка с координатами (*x*, *y*) лежит в четвертой координатной четверти».

**Вариант 36**

1) *y =*  2) *y =*

3) Дана площадь *S* круга. Найти его диаметр *D* и длину *L* окружности, ограничивающей этот круг, учитывая, что *L* = π·*D*,    *S* = π·*D*2/4. В качестве значения π использовать 3.14.

4) Из трехзначного числа *x* вычли его последнюю цифру. Когда результат разделили на 10, а к частному слева приписали последнюю цифру числа *x*, то получилось число *n*. Найти число *n*. По заданному *n* найти число *x* (значение *n* вводится с клавиатуры, 10 *≤ n ≤* 999 и при этом число десятков в *n* не равно нулю).

5) Даны числа *x*, *y*. Проверить истинность высказывания: «Точка с координатами (*x*, *y*) лежит во второй или третьей координатной четверти».

**Вариант 37**

1) *y =*  2) *y =*

3) Даны три точки *A*, *B*, *C* на числовой оси. Найти длины отрезков *AC*   
и *BC* и их сумму.

4) В трехзначном числе *x* зачеркнули первую цифру. Когда оставшееся число  
умножили на 10, а произведение сложили с первой цифрой числа *x*, то получилось число 564. Найти число *x*.

5) Даны числа *x*, *y*. Проверить истинность высказывания: «Точка с координатами (*x*, *y*) лежит в первой или третьей координатной четверти».

**Вариант 38**

1. *y =*  2) *y =*

3) Даны координаты двух противоположных вершин прямоугольника: (*x*1, *y*1), (*x*2, *y*2). Стороны прямоугольника параллельны осям координат. Найти периметр и площадь данного прямоугольника.

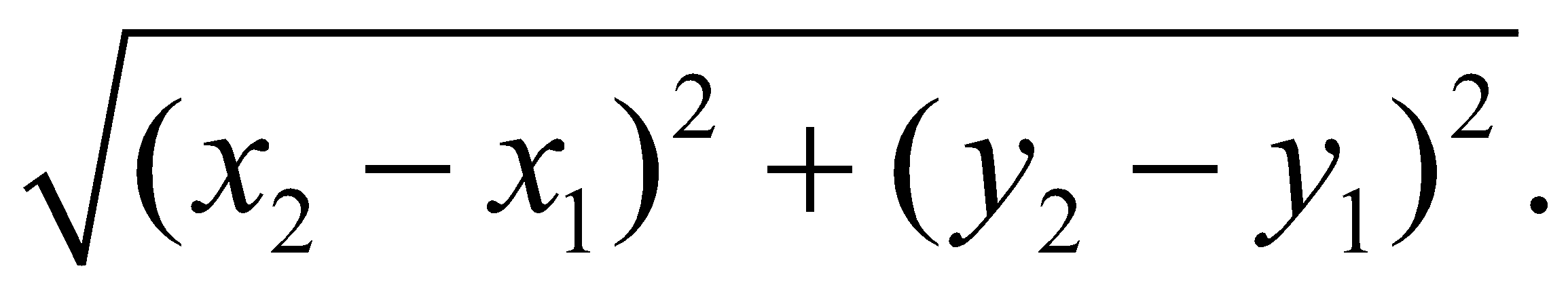
4) С начала 1990 года по некоторый день прошло *n* месяцев и 2 дня. Присвоить  
целочисленной величине *x* значение 1, 2, ..., 11 или 12 в зависимости от того,  
каким месяцем (январем, февралем и т. п.) является месяц этого дня. Например,  
при n=3 значение *х* равно 4.

5) Даны числа *x*, *y*, *x*1, *y*1, *x*2, *y*2. Проверить истинность высказывания: «Точка с координатами (*x*, *y*) лежит внутри прямоугольника, левая верхняя вершина которого имеет координаты (*x*1, *y*1), правая нижняя — (*x*2, *y*2), а стороны параллельны координатным осям».

**Вариант 39**

1. *y* = √*x* − (1 + *x*)·arctg √*x*  2) *y* = *x*·(sin − cos ln *x*)

3) Найти расстояние между двумя точками с заданными координатами (*x*1, *y*1) и (*x*2, *y*2) на плоскости. Расстояние вычисляется по формуле



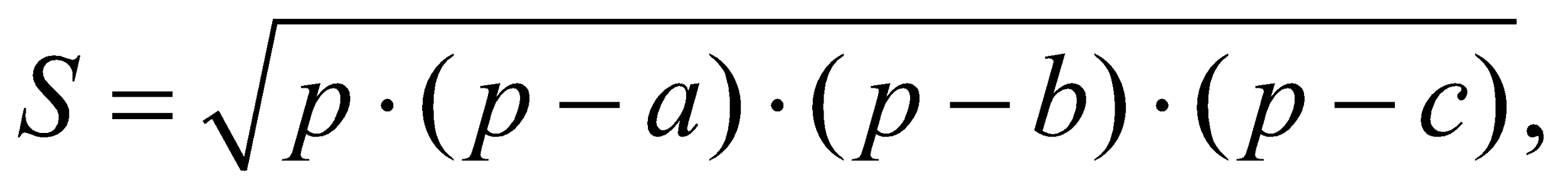
4) С некоторого момента прошло 234 дня. Сколько полных недель прошло за этот период?

5) Даны целые числа *a*, *b*, *c*, являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами *a*, *b*, *c* является равносторонним».

**Вариант 40**

1. *y* = arctg (sh *x*) + (sh *x*)·ln ch *x*  2) *y* = *x*·(sin ln *x* − cos ln *x*)

3) Даны координаты трех вершин треугольника: (*x*1, *y*1), (*x*2, *y*2), (*x*3, *y*3). Найти его периметр и площадь, используя формулу для расстояния между двумя точками на плоскости. Для нахождения площади треугольника со сторонами *a*, *b*, *c* использовать *формулу Герона*:



где *p* = (*a* + *b* + *c*)/2 — *полупериметр*.

4) Дан прямоугольник с размерами 543 130 мм. Сколько квадратов со стороной 130 мм можно отрезать от него?

5) Даны целые числа *a*, *b*, *c*, являющиеся сторонами некоторого треугольника. Проверить истинность высказывания: «Треугольник со сторонами *a*, *b*, *c* является равнобедренным».

# **Условный оператор: группа If**

1. Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае не изменять его. Вывести полученное число.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A: Integer;  **begin**  ReadLn(A);  **if** A > 0 **Then**  A := A + 1;  WriteLn(A);  **end**. | #include <iostream>  using namespace std;  int main(){  int a;  cin>>a;  if (a>0)  a++;  cout<<a<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  if (a>0):  a=a+1  print(a) | static void Main(string[] args){  int a;  a = Convert.ToInt32(Console.ReadLine());  if (a>0)  a = a + 1;  Console.WriteLine (a);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  int A;  Scanner sc=new Scanner(System.in);  A=sc.nextInt();  if(A>0)  A=A+1;  System.out.println(A);  } | | package main  import "fmt"  func main() {  var a int  fmt.Scanf("%d", &a)  if a>0{  a+=1  }  fmt.Printf("%d", a)  } |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a=reader.nextInt()  **if** (a>0)  a++  *println*(**"a=$**a**"**) } | | |
| VB.NET | | |
| Sub Main()  Dim a As Integer  a = Convert.ToInt32(Console.ReadLine())  If a > 0 Then a = a + 1  Console.WriteLine(a)  Console.ReadKey()  End Sub | | |

1. Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае вычесть из него 2. Вывести полученное число.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A: Integer;  **begin**  ReadLn(A);  **if** A > 0 **Then**  WriteLn(A + 1)  **Else**  WriteLn(A - 2);  **end**. | #include <iostream>  using namespace std;  int main(){  int a;  cin>>a;  if (a>0)  a++;  cout<<a<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  if (a>0):  a=a+1  else:  a=a-2  print(a) | static void Main(string[] args){  int a;  a = Convert.ToInt32(Console.ReadLine());  if (a > 0)  a = a + 1;  else  a = a - 2;  Console.WriteLine (a);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  int A;  Scanner sc=new Scanner(System.in);  A=sc.nextInt();  if(A>0)  A=A+1;  else  A=A-2;  System.out.println(A);  } | | package main  import "fmt"  func main() {  var a int  fmt.Scanf("%d", &a)  if a>0{  a+=1  }else{  a=a-2  }  fmt.Println("a=", a)  } |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a=reader.nextInt()  **if** (a>0)  a++  **else** a-=2  *println*(**"a=$**a**"**) } | | |
| VB.NET | | |
| Sub Main()  Dim a As Integer  a = Convert.ToInt32(Console.ReadLine())  If a > 0 Then  a = a + 1  Else  a = a - 2  End If  Console.WriteLine(a)  Console.ReadKey()  End Sub | | |

1. Дано целое число. Если оно является положительным, то прибавить к нему 1; если отрицательным, то вычесть из него 2; если нулевым, то заменить его на 10. Вывести полученное число.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A: Integer;  **begin**  ReadLn(A);  **if** A > 0 **Then**  WriteLn(A + 1)  **Else**  **if** A < 0 **Then**  WriteLn(A - 2)  **Else**  WriteLn(10);  **end**. | #include <iostream>  using namespace std;  int main(){  int a; cin>>a;  if (a>0)  a++;  else if (a<0)  a-=2;  else  a=10;  cout<<a<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  if (a>0):  a=a+1  elif (a<0):  a=a-2  else:  a=10  print(a) | static void Main(string[] args){  int a=Convert.ToInt32(Console.ReadLine());  if (a > 0)  a++;  else if (a < 0)  a = a - 2;  else  a = 10;  Console.WriteLine (a);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  int A;  Scanner sc=new Scanner(System.in);  A=sc.nextInt();  if(A>0){  A=A+1;}  else if (A<0)  A=A-2;  else  A=10;  System.out.println(A);  } | | package main  import "fmt"  func main() {  var a int  fmt.Scanf("%d", &a)  if a>0{  a+=1  }else if a<0{  a=a-2  }else{  a=10  }  fmt.Println("a=", a)  } |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a=reader.nextInt()  **if** (a>0)  a++  **else if** (a<0)  a-=2  **else** a=10  *println*(**"a=$**a**"**) } | | |
| VB.NET | | |
| Sub Main()  Dim a As Integer  a = Convert.ToInt32(Console.ReadLine())  If a > 0 Then  a = a + 1  ElseIf a < 0 Then  a = a - 2  Else  a = 10  End If  Console.WriteLine(a)  Console.ReadKey()  End Sub | | |

1. Даны три целых числа. Найти количество положительных чисел в исходном наборе.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B, C, k: Integer;  **begin**  k := 0;  WriteLn('A,B, C: ');  ReadLn(A, B, C);  **if** A > 0 **Then**  k := k + 1;  **if** B > 0 **Then**  k := k + 1;  **if** C > 0 **Then**  k := k + 1;  WriteLn('k:', k);  **end**. | | #include <iostream>  using namespace std;  int main()  {  int A, B, C, k = 0;  cout << "A, B, C: ";  cin >> A >> B >> C;  if(A > 0) k++;  if(B > 0) k++;  if(C > 0) k++;  cout << "k: " << k << endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("->"))  b= int(input("->"))  c= int(input("->"))  k=0;  if (a>0):  k+=1  if (b>0):  k+=1  if (c>0):  k+=1  print(k) | static void Main(string[] args){  int a, b, c, k=0;  a=Convert.ToInt32(Console.ReadLine());  b=Convert.ToInt32(Console.ReadLine());  c=Convert.ToInt32(Console.ReadLine());  if (a > 0)  k++;  if (b > 0)  k++;  if (c>0)  k++;  Console.WriteLine (k);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int a,b,c,k=0;  Scanner sc=new Scanner (System.in);  a=sc.nextInt();  b=sc.nextInt();  c=sc.nextInt();  if(a>0)  k++;  if(b>0)  k++;  if(c>0)  k++;  System.out.println(k);  } | | | package main  import "fmt"  func main() {  fmt.Println("a,b,c->")  var a,b,c int  var k int  fmt.Scanf("%d %d %d",  &a,&b,&c)  if(a>0){  k+=1  }  if(b>0){  k++  }  if(c>0){  k++}  fmt.Println("K=",k)  } |

1. Даны два числа. Вывести большее из них.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A, B: Integer;  **begin**  WriteLn('A, B: ');  ReadLn(A, B);  **if** A > B **Then**  WriteLn(A)  **Else**  WriteLn(B);  **end**. | Варианти 1.  #include <iostream>  using namespace std;  int main()  {  int a,b;  cin>>a>>b;  if(a > b)  cout<<a<<endl;  else  cout<<b<<endl;  system("pause");  return 0;  }  //Варианти 2.  #include <iostream>  using namespace std;  int main(){  int a,b,mx;  cin>>a>>b;  mx=a>b?a:b;  cout<<mx<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  b= int(input("->"))  if (a>b):  print(a)  else:  print(b) | static void Main(string[] args){  int a, b, mx;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  if (a > b) mx = a;  else  mx = b;  Console.WriteLine (mx);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  int A,B;  Scanner sc=new Scanner(System.in);  A=sc.nextInt();  B=sc.nextInt();  if(A>B)  System.out.println(A);  else  System.out.println(B);  } | | package main  import "fmt"  func main() {  fmt.Println("a,b->")  var a,b,mx int  fmt.Scanf("%d %d",  &a,&b)  if a > b {  mx = a;  }  if a<b {  mx = b; }  fmt.Println("Max=",mx)  } |

1. Даны две переменные вещественного типа: *A*, *B*. Перераспределить значения данных переменных так, чтобы в *A* оказалось меньшее из значений, а в *B* — большее. Вывести новые значения переменных *A* и *B*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B: Real;  **begin**  WriteLn('A, B: ');  ReadLn(A, B);  **if** A < B **Then**  **begin**  (A,B):=(B,A)  **end**;  WriteLn('A=', A, ' B=', B);  **end**. | | #include <iostream>  using namespace std;  int main(){  int a,b,h;  cin>>a>>b;  if (a<b){  h=a;  a=b;  b=h;  }  cout<<a<<" "<<b<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("->"))  b= int(input("->"))  if (a>b):  (a,b)=(b,a)  print(a,' ',b) | static void Main(string[] args){  int a, b, h;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  if(a>b){  c=b;  b=a;  a=c;  }  Console.WriteLine(a+" "+b);  } | | |
| Java | | | Go |
| public static void main(String[]args){  int a,b,c;  Scanner sc=new Scanner (System.in);  a=sc.nextInt();  b=sc.nextInt();  if(a>b){  c=b;  b=a;  a=c;  }  System.out.println(a+" "+b);  } | | | package main  import "fmt"  func main() {  var a,b,c int  fmt.Scanf("%d %d",&a,&b)  if(a>b){  c=b;  b=a;  a=c;}  fmt.Println("a=",a,"\nb=",b)  } |

1. Даны три числа. Найти наименьшее из них.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B, C, mn: Real;  **begin**  WriteLn('A,B, C: ');  ReadLn(A, B, C);  Mn := a;  **if** mn < B **Then**  Mn := B;  **if** mn > C **then**  Mn := c;  WriteLn('Min = ', Mn)  **end**. | | #include <iostream>  using namespace std;  int main()  {  int a,b,c,mn;  cin>>a>>b>>c;  mn=a;  if (mn<b)  mn=b;  if (mn<c)  mn=c;  cout<<mn<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("->"))  b= int(input("->"))  c= int(input("->"))  mn=a  if (mn>b):  mn=b  if (mn>c):  mn=c;  print(mn) | static void Main(string[] args)  {  int a, b, c, mn;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  c = Convert.ToInt32(Console.ReadLine());  mn=a;  if (mn<b)  mn=b;  if (mn<c)  mn=c;  Console.WriteLine(mn);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int a,b,c,d;  Scanner sc=new Scanner (System.in);  a=sc.nextInt();  b=sc.nextInt();  c=sc.nextInt();  d=a;  if(d>b)  c=b;  if (d>c)  d=c;  System.out.println(d);  } | | | package main  import "fmt"  func main() {  var a,b,c,mn int  fmt.Scanf("%d %d %d",  &a,&b,&c)  mn=a;  if mn>b{  mn=b;  }  if mn>c {  mn=c;  }  fmt.Println("Min=",mn)  } |

1. Даны три переменные вещественного типа: . Если их значения упорядочены по возрастанию, то удвоить их; в противном случае заменить значение каждой переменной на противоположное. Вывести новые значения переменных .

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B, C: Real;  **begin**  WriteLn('A,B,C:');  ReadLn(A, B, C);  **if** (A<B) **And** (B<C) **Then**  **begin**  A := A \* 2;  B := B \* 2;  C := C \* 2;  **end**  **Else**  **begin**  A := -A;  B := -B;  C := -C;  **end**;  WriteLn('A=', A, ' B=', B,  ' C = ', C);  **end**. | | #include <iostream>  using namespace std;  int main(){  int a,b,c;  cin>>a>>b>>c;  if ((a < b)&&(b < c)) {  a\*=2;  b\*=2;  c\*=2; }  else {  a\*=-1;  b\*=-1;  c\*=-1;  }  cout<<a<<" "<<b<<" "<<c<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("->"))  b= int(input("->"))  c= int(input("->"))  if (a<b<c):  a=a\*2  b=b\*2  c=c\*2  else:  a=-a  b=-b  c=-c  print(a,' ',b,' ',c) | static void Main(string[] args){  int a, b, c;  a=Convert.ToInt32(Console.ReadLine( ));  b=Convert.ToInt32(Console.ReadLine( ));  c=Convert.ToInt32(Console.ReadLine( ));  if ((a < b) && (b < c))  {  a = a \* 2;  b = b \* 2;  c = c \* 2;  }  else{  a=-a;  b=-b;  c=-c;  }  Console.WriteLine(a);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int a,b,c;  Scanner sc=new Scanner (System.in);  a=sc.nextInt();  b=sc.nextInt();  c=sc.nextInt();  if((a<b)&&(b<c)){  a=a\*2;  b=b\*2;  c=c\*2;  }  else{  a=-a;  b=-b;  c=-c;  }  System.out.println(a+" "+b+" "+c); | | | package main  import "fmt"  func main() {  var a,b,c int  fmt.Scanf("%d %d %d",  &a,&b,&c)  if((a<b)&&(b<c)){  a=a\*2;  b=b\*2;  c=c\*2;  }else{  a=-a;  b=-b;  c=-c;  }  fmt.Println("a=",a,"b=",b,  "c=",c)  } |

1. Даны три целых числа, одно из которых отлично от двух других, равных между собой. Определить порядковый номер числа, отличного от остальных.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **var**  A, B, C: Real;  **begin**  Readln(A, B, C);    **if** B = C **then**  Writeln('1');  **if** A = C **then**  Writeln('2');  **if** A = B **then**  Writeln('3');  **end**. | #include <iostream>  using namespace std;  int main(){  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  int a, b, c, n;  cin >> a>>b>>c;  if (b == c)  n = 1;  else if (a == c )  n = 2;  else if (a == b )  n = 3;  cout << n << endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  b= int(input("->"))  c= int(input("->"))  k=0  if (b==c):  k=1;  if (a==c):  k=2  if (a==b):  k=3  print(k) | static void Main(string[] args){  int a, b, c;  a=Convert.ToInt32(Console.ReadLine( ));  b=Convert.ToInt32(Console.ReadLine( ));  c=Convert.ToInt32(Console.ReadLine( ));  if ((a==b) && (b!=c))  Console.WriteLine(3);  elseif ((a == c) && (c != b))  Console.WriteLine(2);  else if ((c == b) && (b != a))  Console.WriteLine(1);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[] args) {  int a,b,c,h = 0;  Scanner sc=new Scanner (System.in);  a=sc.nextInt();  b=sc.nextInt();  c=sc.nextInt();  if((a==b)&&(b!=c)){  h=3;  }  if((a==c)&&(c!=b)){  h=2;  }  if((b==c)&&(c!=a)){  h=1;  }  System.out.println(h); | | package main  import ("fmt"  )  func main() {  var a,b,c int  fmt.Scanf("%d %d %d",  &a,&b,&c)  if((a==b)&&(b!=c)){  fmt.Println("N=",3)  }  if((a==c)&&(c!=b)){  fmt.Println("N=",2)  }  if((b==c)&&(c!=a)){  fmt.Println("N=",1)  }  } |

1. На числовой оси расположены три точки: *A*, *B*, *C*. Определить, какая из двух последних точек (*B* или *C*) расположена ближе к *A*, и вывести эту точку и ее расстояние от точки *A*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B, C: Real;  **begin**  WriteLn('A,B, C: ');  ReadLn(A, B, C);  **if** (B - A) < (C - A) **Then**  WriteLn('B: ', (B - A))  **Else**  WriteLn('C: ', (C - A));  **end**. | | #include <iostream>  using namespace std;  int main(){  int c,b,a;  cout << "a,b,c" ;  cin >>a>>b>>c;  if ( abs(b-a)>abs(c-a))  cout<<"c-a:="<<abs(c-a);  else  cout<<"b-a:="<<abs(b-a);  system("pause");  return 0;  } | |
| Python | C# | | |
| from math import sqrt  a= int(input("->"))  b= int(input("->"))  c= int(input("->"))  x=abs(b-a)  y=abs(c-a)  if (x<y):  print('b:',x)  elif (x>y):  print('c:',y)  else:  print('AB=AC') | static void Main(string[] args){  int a, b, c, x, y;  a=Convert.ToInt32(Console.ReadLine());  b=Convert.ToInt32(Console.ReadLine());  c=Convert.ToInt32(Console.ReadLine());  x= Math.abs(b-a);  y = Math.abs(c - a);  if (x<y)  Console.WriteLine(b);  else  Console.WriteLine(a);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  double a,b,c,s;  Scanner sc=new Scanner (System.in);  a=sc.nextDouble();  b=sc.nextDouble();  c=sc.nextDouble();  if(b<c){  s=Math.abs(a-b);  System.out.println(b+" "+s);}  if(c<b){  s=Math.abs(a-c);  System.out.println(c+" "+s);  } | | | package main  import ("fmt"  "math")  func main() {  var a,b,c,s float64  fmt.Scanf("%v %v %v",  &a,&b,&c)  if(b<c){  s=math.Abs(a-b);  fmt.Println("B=",s)  }  if(c<b){  s=math.Abs(a-c);  fmt.Println("C=",s)  }  } |

1. Даны целочисленные координаты трех вершин прямоугольника, стороны которого параллельны координатным осям. Найти координаты его четвертой вершины.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  x1, y1, x2, y2, x3, y3, x4, y4: integer;  **begin**  ReadLn(x1, y1, x2, y2, x3, y3);  **if** x1 = x2 **Then**  x4 := x3;  **if** x1 = x3 **Then**  x4 := x2;  **if** x2 = x3 **Then**  x4 := x1;  **if** y1 = y2 **Then**  y4 := y3;  **if** y1 = y3 **Then**  y4 := y2;  **if** y2 = y3 **Then**  y4 := y1;  WriteLn('(x4, y4: ',  x4, ',', y4);  **end**. | | #include <iostream>  using namespace std;  int main(){  int x1,y1,x2,y2,x3,y3,x4,y4;  cin>> x1>>y1>>x2>>y2>>x3>>y3;  if (x1==x2) x4=x3;  if (x3==x2) x4=x1;  if (x1==x3) x4=x2;  if (y1==y2) y4=y3;  if (y3==y2) y4=y1;  if (y1==y3) y4=y2;  cout<<x4<<" "<<y4<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| x1= int(input("->"))  y1=int(input("->"))  x2= int(input("->"))  y2=int(input("->"))  x3= int(input("->"))  y3=int(input("->"))  if (x1==x2):  x4=x3;  if (x3==x2):  x4=x1  if (x1==x3):  x4=x2  if (y1==y2):  y4=y3;  if (y3==y2):  y4=y1  if (y1==y3):  y4=y2  print(x4,' ',y4); | static void Main(string[] args){  int x1,y1,x2,y2,x3,y3,x4,y4;  x1=Convert.ToInt32(Console.ReadLine( ));  y1=Convert.ToInt32(Console.ReadLine( ));  x2=Convert.ToInt32(Console.ReadLine( ));  y2=Convert.ToInt32(Console.ReadLine( ));  x3=Convert.ToInt32(Console.ReadLine( ));  y3=Convert.ToInt32(Console.ReadLine( ));  if (x1==x2) x4=x3;  if (x3==x2) x4=x1;  if (x1==x3) x4=x2;  if (y1==y2) y4=y3;  if (y3==y2) y4=y1;  if (y1==y3) y4=y2;  Console.WriteLine(x4+” “+y4);  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int x1,y1,x2,y2,x3,y3,x4,y4;  Scanner sc=new Scanner  (System.in);  x1=sc.nextInt();  y1=sc.nextInt();  x2=sc.nextInt();  y2=sc.nextInt();  x3=sc.nextInt();  y3=sc.nextInt();  if (x1==x2) x4=x3;  if (x3==x2) x4=x1;  if (x1==x3) x4=x2;  if (y1==y2) y4=y3;  if (y3==y2) y4=y1;  if (y1==y3) y4=y2;  System.out.println(x4+” “+y4);  } | | | package main  import "fmt"  func main() {  fmt.Println("x1,y1,x2,y2,x3, y3")  var x1,y1,x2,y2,x3,y3,x4,y4 int  fmt.Scanf("%d %d %d %d %d %d",  &x1,&y1,&x2,&y2,&x3,&y3)  if (x1==x2) {x4=x3};  if (x3==x2) {x4=x1;}  if (x1==x3) {x4=x2;}  if (y1==y2) {y4=y3;}  if (y3==y2) {y4=y1;}  if (y1==y3) {y4=y2;}  fmt.Println("x4=",x4,"\ny4=",y4)  } |

1. Для данного вещественного *x* найти значение следующей функции , принимающей вещественные значения:

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  x, y: Real;  **begin**  ReadLn(x);  **if** x <= 0 **Then**  y := -x  **Else**  **if** (x>0) **and** (x<2) **Then**  y := Sqr(x)  **Else**  y := 4;  WriteLn('f(x)=', y);  **end**. | | #include <cmath>  #include <iostream>  using namespace std;  int main(){  float x,y;  cin>> x;  if (x<=0)  y=-x;  else if ((0<x) && (x<=2))  y=4;  else  y=sqrt(x);  cout<<y<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| x= float(input("->"))  if (x<=0):  y=-x  elif ((0<x) and (x<2)):  y=x\*\*2  else:  y=4  print(y); | static void Main(string[] args){  int y, x;  x = Convert.ToInt32(Console.ReadLine());  if(x<=0)  f=-x;  else if((0<x) && (x<2))  f=x\*x;  else  f=4;  Console.WriteLine(f);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Double x,f = null;  Scanner sc=new Scanner(System.in);  x=sc.nextDouble();  if(x<=0)  f=-x;  else if((0<x) && (x<2))  f=x\*x;  else  f=4;  System.out.println(f);  } | | | package main  import ("fmt"  )  func main() {  var x,y int  fmt.Scanf("%d",&x)  if x<=0 {  y=-x  }else if x>0 && x<2 {  y=x\*x  }else{  y=4}  fmt.Println("F=",y)  } |

1. Дан номер года (положительное целое число). Определить количество дней в этом году, учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400 (например, годы 300, 1300 и 1900 не являются високосными, а 1200 и 2000 — являются).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  Year: Integer;  **begin**  ReadLn(Year);  **if** Year **Mod** 4 = 0 **Then**  **begin**  **if** (Year **Mod** 100=0) **And**  (Year **Mod** 400<>0) **Then**  **begin**  WriteLn('365')  **end**  **Else**  WriteLn('366');  **end**  **Else**  WriteLn('365');  **end**. | | #include <iostream>  using namespace std;  int main(){  int y;  cin >>y;  if (y % 4==0)  {  if ((y % 100==0)&& (y % 400)!=0))  cout << "365"<<<endl;  else  cout << "366"<<endl;  }  else  cout << "365"<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("->"))  if (n%4==0):  if((n%100==0) and  (n%400!=0)):  print(365)  else:  print(366)  else:  print(365); | static void Main(string[] args){  int n;  n = Convert.ToInt32(Console.ReadLine());  if (n % 4 == 0)  if ((n % 100 == 0) && (n % 400 != 0))  Console.WriteLine(365);  else  Console.WriteLine(366);  else  Console.WriteLine(365);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int y;  Scanner sc=new Scanner(System.in);  y=sc.nextInt();  if (y % 4==0){  if ((y%100==0)&& (y % 400)!=0))  System.out.println("365");  else  System.out.println("366");  }  else  System.out.println("365");  } | | | package main  import ("fmt"  )  func main() {  var y int  fmt.Scanf("%d",&y)  if (y % 4==0){  if((y%100==0)&&(y%400)!=0){  fmt.Println("365");  }else {  fmt.Println("366");  }}else{  fmt.Println("365")  }  } |

# **Оператор выбора: группа Case**

1. Дано целое число в диапазоне 1–7. Вывести строку — название дня недели, соответствующее данному числу (1 — «понедельник», 2 — «вторник» и т. д.).

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Var**  x: Integer;  **begin**  ReadLn(x);  **case** x **Of**  1: WriteLn('Monday');  2: WriteLn('Tuesday');  3: WriteLn('Wednesday');  4: WriteLn('Thursday');  5: WriteLn('Friday');  6: WriteLn('Shanbe');  7: WriteLn('Sunday')  **else**  Writeln('Error');  **End**;  **end**. | | #include <iostream>  using namespace std;  int main(){  int x;  cin >> x;  switch (x){  case 1:cout << "Monday" << endl;  break;  case 2:cout << "Tuesday" << endl;  break;  case 3:cout<<"Wednesday "<< endl;  break;  case 4:cout <<"Thursday"<<endl;  break;  case 5:cout << "Friday" << endl;  break;  case 6:cout << "Shanbe" << endl;  break;  case 7:cout <<"Sunday"<<endl;  break;  default:cout<<"Error";  }  system("pause");  return 0;  } |
| Python | C# | |
| def switch\_case(case):  return {  1 : "Monday",  2 : "Tuesday",  3 : "Wednesday ",  4: "Thursday",  5 : "Friday",  6 : "Saturday",  7 : "Sunday"  }.get(case, "Error")  n= int(input("->"))  print(switch\_case(n)) | static void Main(string[] args){  int n;  String S="";  n = Convert.ToInt32(Console.ReadLine());  switch (n){  case 1: S="Monday"; break;  case 2: S="Tuesday"; break;  case 3: S="Wednesday ";break;  case 4: S="Thursday";break;  case 5: S="Friday"; break;  case 6: S="Shanbe"; break;  case 7: S="Sunday";break;  default: S="Error");break;  }  Console.WriteLine(S);  } | |
| Java | | |
| public static void main(String[] args) {  int n;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  switch(n){  case 1:System.out.println("Monday");break;  case 2:System.out.println("Tuesday");break;  case 3:System.out.println("Wednesday ");break;  case 4:System.out.println("Thursday");break;  case 5:System.out.println("Friday");break;  case 6:System.out.println("Shanbe");break;  case 7:System.out.println("Sunday");break;  default:System.out.println("Error!");break;  } | | |
| Go | | |
| package main  import "fmt"  func main()  {  fmt.Println("a=?")  var a int  fmt.Scanf("%d", &a)  switch a {  case 1:  fmt.Println("Monday")  case 2:  fmt.Println("Tuesday")  case 3:  fmt.Println("Wednesday ")  case 4:  fmt.Println("Thursday")  case 5:  fmt.Println("Friday")  case 6:  fmt.Println("Shanbe")  case 7:  fmt.Println("Sunday")  default:  fmt.Println("Error")  }  } | | |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>)  {  **val** reader = Scanner(System.*`in`*)  **var** n=reader.nextInt()  **var** S:String  **when**(n) {  1 -> S = **"Monday"** 2 -> S = **"Tuesday"** 3 -> S = **"Wednesday "** 4 -> S = **"Thursday"** 5 -> S = **"Friday"**;  6 -> S = **"Shanbe"** 7 -> S = **"Sunday"  else**-> S = **"Error"** }  *println*(S) } | | |
| VB.NET | | |
| Sub Main()  Dim a As Integer  Dim S As String  a = Convert.ToInt32(Console.ReadLine())  Select Case a  Case 1 : S = "Monday"  Case 2 : S = "Tuesday"  Case 3 : S = "Wednesday "  Case 4 : S = "Thursday"  Case 5 : S = "Friday"  Case 6 : S = "Shanbe"  Case 7 : S = "Sunday"  Case Else : S = "Error"  End Select  Console.WriteLine(S)  Console.ReadKey()  End Sub | | |

1. Дан номер месяца — целое число в диапазоне 1–12 (1 — январь, 2 — февраль и т. д.). Вывести название соответствующего времени года («зима», «весна», «лето», «осень»).

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Var**  M: Integer;  **begin**  Write(' month: ');  Readln(M);  **case** M **of**  12, 1, 2: Writeln('Zima');  3..5: Writeln('Vesna');  6..8: Writeln('Leto');  9..11: Writeln('Osen');  **end**;  **end**. | | #include <iostream>  using namespace std;  int main(){  int month;  cin>> month;  switch (month) {  case 12:case 1:case 2:  cout<<"Zima\n" ;  break;  case 3:case 4:case 5:  cout<<"Vesna\n" ;  break;  case 6:case 7:case 8:  cout<<"Leto\n" ;  break;  case 9:case 10:case 11:  cout<<"Osen\n" ;  break;  }  system("pause");  return 0;  } |
| Python | C# | |
| def switch\_case(case):  return {  12:"Zima",  1:"Zima",  2:"Zima",  3:"Vesna",  4: "Vesna",  5: "Vesna",  6:"Leto",  7:"Leto",  8:"Leto",  9:"Osen",  10:"Osen",  11:"Osen",  }.get(case, "an out of range number")  n= int(input("->"))  print(switch\_case(n)) | static void Main(string[] args){  int n;  n= Convert.ToInt32(Console.ReadLine());  switch(n){  case 12:  case 1:  case 2:Console.WriteLine("Zima");  break;  case 3:  case 4:  case 5:Console.WriteLine ("Vesna");  break;  case 6:  case 7:  case 8:Console.WriteLine("Leto");  break;  case 9:  case 10:  case 11:Console.WriteLine ("Osen");  break;  default:Console.WriteLine ("Error");  break;  }  Console.ReadKey ();  } | |
| Java | | |
| public static void main(String[] args) {  int k;  Scanner sc=new Scanner(System.in);  k=sc.nextInt();  switch(k){  case 12:  case 1:  case 2:System.out.println("Zima");break;  case 3:case 4:  case 5:System.out.println("Vesna");break;  case 6:case 7:  case 8:System.out.println("Leto");break;  case 9:case 10:  case 11:System.out.println("Osen");break;  default:System.out.println("Error!");break;  } | | |
| Kotlin | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** n=reader.nextInt()  **var** S=**""  when**(n) {  12,1,2 -> S = **"Zima"** 3, 4,5 -> S = **"Vesna"  in** 6..8 -> S = **"Leto "** 9,10,11 -> S = **"Osen"  !in** 1..12 -> S = **"Error"** }  *println*(S) } | | |

1. Арифметические действия над числами пронумерованы следующим образом: 1 — сложение, 2 — вычитание, 3 — умножение, 4 — деление. Дан номер действия *N* (целое число в диапазоне 1–4) и вещественные числа *A* и *B* (*В* не равно 0). Выполнить над числами указанное действие и вывести результат.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  N, A, B: Integer;  **begin**  Write('N,A,B: ');  Readln(N, A, B);    **case** N **of**  1: Writeln(A + B);  2: Writeln(A - B);  3: Writeln(A \* B);  4: Writeln(A / B);  **end**;  **end**. | #include <iostream>  using namespace std;  int main()  {  int n;  float a,b;  cin>>n>>a>>b;  switch (n) {  case 1:  cout<<a+b<<endl;  break;  case 2:  cout<<a-b<<endl;  break;  case 3:  cout<<a\*b<<endl;  break;  case 4:  cout<<a/b<<endl;  break;  }  system("pause");  return 0;  } |
| C# | |
| static void Main(string[] args){  int n;  double a, b;  n = Convert.ToInt32(Console.ReadLine());  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  switch(n)  {  case 1:Console.WriteLine(a+b);break;  case 2: Console.WriteLine(a-b); break;  case 3: Console.WriteLine(a\*b); break;  case 4: Console.WriteLine(a/b); break;  default:  Console.WriteLine ("Error"); break;  }  Console.ReadKey ();  } | |
| Java | |
| public static void main(String[] args) {  int k,a,b;  Scanner sc=new Scanner(System.in);  k=sc.nextInt();  a=sc.nextInt();  b=sc.nextInt();  switch(k){  case 1:System.out.println(a+b);break;  case 2:System.out.println(a-b);break;  case 3:System.out.println(a\*b);break;  case 4:System.out.println(a/b);break;  default:System.out.println("Error");break;  } | |

5) Даны два целых числа: *D* (день) и *M* (месяц), определяющие правильную дату невисокосного года. Вывести значения *D* и *M* для даты, предшествующей указанной.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  D, M: Integer;  **begin**  Write('D,M: ');  Readln(D, M);  D := D - 1;  **if** D = 0 **then**  **begin**  M := M - 1;  **if** M =0 **then** M := 12;  **case** M **of**  1,3,5,7,8,10,12:  D:= 31;  4, 6, 9, 11: D := 30;  2: D := 28;  **end**;  **end**;  Writeln('D.M: ', D, '.', M);  **end**. | #include <iostream>  using namespace std;  int main(){  int d, m;  cout << "d,m \n";  cin >> d>>m;  if (d == 1) {  switch (m)  {  case 1: d = 31;break;  case 3: d = 31;break;  case 5: d = 31;break;  case 7: d = 31;break;  case 8: d = 31;break;  case 10: d = 31;break;  case 12: d = 31;break;  case 4: d = 30;break;  case 6: d = 30;break;  case 9: d = 30;break;  case 11: d = 30;break;  case 2: d = 28;break;  }  }  else  d=d-1;  cout<<d<<":"<<m<<endl;  system("pause");  return 0;  } |
| C# | |
| static void Main(string[] args){  int d, m;  d = Convert.ToInt32(Console.ReadLine());  m = Convert.ToInt32(Console.ReadLine());  if (d == 1)  switch (m){  case 1: Console.WriteLine(31 + " " + 12); break;  case 2: Console.WriteLine(31 + " " + 1); break;  case 3: Console.WriteLine(28 + " " + 2); break;  case 4: Console.WriteLine(31 + " " + 3); break;  case 5: Console.WriteLine(30 + " " + 4); break;  case 6: Console.WriteLine(31 + " " + 5); break;  case 7: Console.WriteLine(30 + " " + 6); break;  case 8: Console.WriteLine(31 + " " + 7); break;  case 9: Console.WriteLine(31 + " " + 8); break;  case 10: Console.WriteLine(30 + " " + 9); break;  case 11: Console.WriteLine(31 + " " + 10); break;  case 12: Console.WriteLine(30 + " " + 11); break;  }  else  Console.WriteLine(d-1+ " " + m);  Console.ReadKey();  } | |
| Java | |
| public static void main(String[] args) {  int D,M;  Scanner sc=new Scanner(System.in);  D=sc.nextInt();  M=sc.nextInt();  if(D==1)  switch(M){  case 1:System.out.println(31+" "+12);break;  case 2:System.out.println(31+" "+1);break;  case 3:System.out.println(28+" "+2);break;  case 4:System.out.println(31+" "+3);break;  case 5:System.out.println(30+" "+4);break;  case 6:System.out.println(31+" "+5);break;  case 7:System.out.println(30+" "+6);break;  case 8:System.out.println(31+" "+7);break;  case 9:System.out.println(31+" "+8);break;  case 10:System.out.println(30+" "+9);break;  case 11:System.out.println(31+" "+10);break;  case 12:System.out.println(30+" "+11);break;}  else{  D=D-1;  System.out.println(D+" "+M);  } | |

6) Робот может перемещаться в четырех направлениях («С» — север,   
«З» — запад, «Ю» — юг, «В» — восток) и принимать три цифровые команды: 0 — продолжать движение, 1 — поворот налево, –1 — поворот направо. Дан символ *C* — исходное направление робота и целое число *N* — посланная ему команда. Вывести направление робота после выполнения полученной команды.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  C: Char;  N: Integer;  **begin**  Write('C:');  Readln(C);  Write('N:');  Readln(N);    **case** C **of**  'N':  **case** N **of**  1: C := 'W';  0: C := 'N';  -1: C:='E';  **end**;  'E':  **case** N **of**  1: C := 'N';  0: C := 'E';  -1: C:='S';  **end**;  'S':  **case** N **of**  1: C := 'E';  0: C := 'S';  -1: C:='W';  **end**;  'W':  **case** N **of**  1: C := 'S';  0: C := 'W';  -1: C:='N';  **end**;  **end**;  Writeln(C);  **end**. | #include <iostream>  using namespace std;  int main()  {  int n;  char c;  cin>>c>>n;  if (c=='С')  switch (n) {  case 1: c='З'; break;  case 0:c='С'; break;  case -1: c='В'; break;  }  else if (c=='В')  switch (n) {  case 1: c='С'; break;  case 0:c='В';break;  case -1:c='Ю';break;  }  else if (c=='Ю')  switch (n) {  case 1:c='В'; break;  case 0:c='Ю'; break;  case -1: c='З';break;  }  else if (c=='З')  switch (n) {  case 1: c='Ю';break;  case 0:c='З'; break;  case -1: c='С'; break;  }  cout<<c<<endl;  system("pause");  return 0;  } |
| C# | |
| static void Main(string[] args){  int n;  char S;  S = Convert.ToChar(Console.ReadLine());  n = Convert.ToInt32(Console.ReadLine());  if (S == 's') {  switch (n) {  case 0: Console.WriteLine("Sever"); break;  case 1: Console.WriteLine("Zapad"); break;  case -1: Console.WriteLine("Vostok"); break;  }  }  else if (S == 'z'){  switch (n){  case 0: Console.WriteLine("Zapad"); break;  case 1: Console.WriteLine("Yug"); break;  case -1: Console.WriteLine("Sever"); break;  }  }  else if (S == 'v'){  switch (n){  case 0: Console.WriteLine("Vostok"); break;  case 1: Console.WriteLine("Sever"); break;  case -1: Console.WriteLine("Yug"); break;  }  }  else if (S == 'y')  {  switch (n)  {  case 0: Console.WriteLine("Yug"); break;  case 1: Console.WriteLine("Vostok"); break;  case -1: Console.WriteLine("Zapad"); break;  }  }  Console.ReadKey();  } | |
| Java | |
| public static void main(String[] args) {  int n;  String c, c1=null;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  c= sc.next();  if(c.equals("z"))  switch(n){  case 0: c1="z";break;  case 1: c1="v";break;  case -1: c1="y";break;  }  else if(c.equals("s"))  switch(n){  case 0: c1="s";break;  case 1: c1="z";break;  case -1: c1="y";break;  }  else if(c.equals("y"))  switch(n){  case 0: c1="y";break;  case 1: c1="s";break;  case -1: c1="v";break;  }  else if(c.equals("v"))  switch(n){  case 0: c1="v";break;  case 1: c1="y";break;  case -1: c1="z";break;  }  System.out.println(c1);  } | |

7) Элементы окружности пронумерованы следующим образом: 1 — радиус *R*, 2 — диаметр *D* = 2·*R*, 3 — длина *L* = 2· ·*R*, 4 — площадь круга *S* =  ·*R*2. Дан номер одного из этих элементов и его значение. Вывести значения остальных элементов данной окружности (в том же порядке). В качестве значения использовать 3.14.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  h, R, D, L, S: real;  n: integer;  **begin**  read(n);  read(h);  **case** n **of**  1: R := h;  2: R := h / 2;  3: R:=h/2\*3.14;  4: R:=sqrt(h/3.14);  **end**;  D := 2 \* R;  L := 2 \* 3.14;  S := 3.14 \* R \* R;  Writeln(R, ' ', D, ' ', L, ' ', S);  **end**. | #include <iostream>  using namespace std;  int main()  {  int n;  double h,R=0,D,L,S;  cin>>n>>h;  switch(n){  case 1:R=h;break;  case 2:R=h/2;break;  case 3:R=h/2\*3.14;break;  case 4:R= sqrt(h/3.14);break;  }  D=2\*R;  L=2\*3.14;  S=3.14\*R\*R;  cout<<R<<" "<<D<<" ";  cout<<<<L<<" "<<S<<endl;  system("pause");  return 0;  } |
| C# | |
| static void Main(string[] args){  Double h, D, R, L, S;  int n;  h = Convert.ToDouble(Console.ReadKey());  n = Convert.ToInt32(Console.ReadLine());  switch (n)  {  case 1: R = h; break;  case 2: R = h / 2; break;  case 3: R = h / 2 \* 3.14; break;  case 4: R = Math.Sqrt(h / 3.14); break;  }  D = 2 \* R;  L = 2 \* 3.14 \* R;  S = 3.14 \* (R \* R);  Console.WriteLine("R={0},D={1},L={2},S={3}", R,D,L,S);  Console.ReadKey();  } | |
| Java | |
| public static void main(String[] args) {  int n;  double h,R=0,D,L,S;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  h=sc.nextDouble();  switch(n){  case 1:R=h;break;  case 2:R=h/2;break;  case 3:R=h/2\*Math.PI;break;  case 4:R=Math.sqrt(h/Math.PI);break;  }  D=2\*R;  L=2\*Math.PI\*R;  S=Math.PI\*R\*R;  System.out.println(R+" "+" "+D+" "+L+" "+S+" ");  } | |

1. Мастям игральных карт присвоены порядковые номера: 1 — пики,   
   2 — трефы, 3 — бубны, 4 — червы. Достоинству карт, старших десятки, присвоены номера: 11 — валет, 12 — дама, 13 — король, 14 — туз. Даны два целых числа: *N* — достоинство и *M* — масть карты   
    Вывести название соответствующей карты вида «шестерка бубен», «дама червей», «туз треф» и т. п.

| PascalABC.NET |
| --- |
| **var**  N, M: Integer;  **begin**  Write('Введите достоинство карты: ');  Readln(N);  Write('Введите масть карты: ');  Readln(M);  **case** N **of**  6: write('шестёрка ');  7: write('семёрка ');  8: write('восьмёрка ');  9: write('девятка ');  10: write('десятка ');  11: write('валет ');  12: write('дама ');  13: write('король ');  14: write('туз ');  **end**;  **case** M **of**  1: write('пик');  2: write('треф');  3: write('бубен');  4: write('червей');  **end**;  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main(){  int n,m;  cin>>n>>m;  switch (n) {  case 6: cout<<"шестёрка";break;  case 7: cout<<"семёрка "; break;  case 8: cout<<"восьмёрка "; break;  case 9: cout<<"девятка ";break;  case 10:cout<<"десятка "; break;  case 11:cout<<"валет "; break;  case 12: cout<<"дама ";break;  case 13:cout<<"король ";break;  case 14:cout<<"туз ";break;  }  switch (m) {  case 1:cout<<"пик\n"; break;  case 2:cout<<"треф\n"; break;  case 3:cout<<"бубен\n";  break;  case 4:cout<<"червы\n";  break;  }  system("pause");  return 0;  } |

1. Дано целое число в диапазоне 20–69, определяющее возраст (в годах). Вывести строку-описание указанного возраста, обеспечив правильное согласование числа со словом «год», например: 20 — «двадцать лет», 32 — «тридцать два года», 41 — «сорок один год».

| PascalABC.NET |
| --- |
| **Var**  Y: Integer;  **begin**  Write('Введите возраст: ');  Readln(Y);  **case** Y **div** 10 **of**  2: write('двадцать ');  3: write('тридцать ');  4: write('сорок ');  5: write('пятьдесят ');  6: write('шестьдесят ');  **end**;  **case** Y **mod** 10 **of**  1: write('один ');  2: write('два ');  3: write('три ');  4: write('четыре ');  5: write('пять ');  6: write('шесть ');  7: write('семь ');  8: write('восемь ');  9: write('девять ');  **end**;  **case** Y **mod** 10 **of**  0, 5, 6, 7, 8, 9: write('лет.');  1: write('год.');  2, 3, 4: write('года.');  **end**;  **end**. |
| C++ |
| #include <iostream>  #include <windows.h>  using namespace std;  int main()  {  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  int y;  cin >> y;  switch (y / 10) {  case 2: cout << "двадцать "; break;  case 3: cout << "тридцать "; break;  case 4: cout << "сорок "; break;  case 5: cout << "пятдесят "; break;  case 6: cout << "шестьдесят "; break;  }  switch (y % 10)  {  case 1: cout << "один "; break;  case 2: cout << "два "; break;  case 3: cout << "три "; break;  case 4: cout << "четыре "; break;  case 5: cout << "пять "; break;  case 6: cout << "шесть "; break;  case 7: cout << "семь "; break;  case 8: cout << "восемь "; break;  case 9: cout << "девять "; break;  }  switch (y % 10)  {  case 0:  case 5:  case 6:  case 7:  case 8:  case 9:  cout << "лет.\n "; break;  case 1:cout << "год.\n "; break;  case 2:  case 3:  case 4:  cout << "года.\n "; break;  }  system("pause");  return 0;  } |

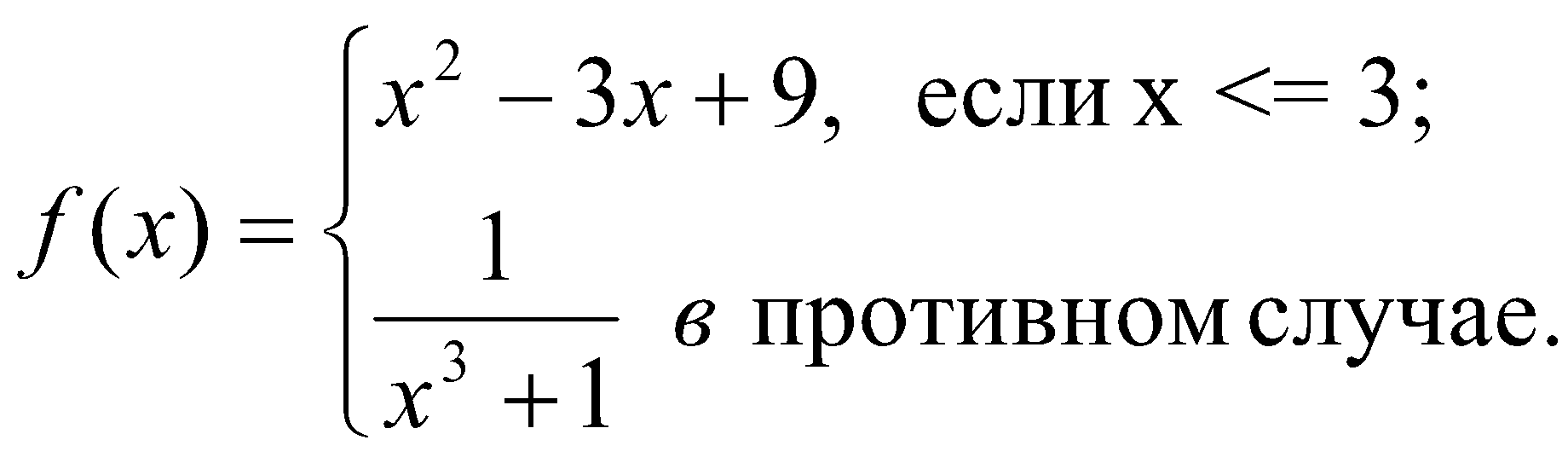
1. Даны два целых числа: *D* (день) и *M* (месяц), определяющие правильную дату. Вывести знак Зодиака, соответствующий этой дате: «Водолей» (20.1–18.2), «Рыбы» (19.2–20.3), «Овен» (21.3–19.4), «Телец» (20.4–20.5), «Близнецы» (21.5–21.6), «Рак» (22.6–22.7), «Лев» (23.7–22.8), «Дева» (23.8–22.9), «Весы» (23.9–22.10), «Скорпион» (23.10–22.11), «Стрелец» (23.11–21.12), «Козерог» (22.12–19.1).

| PascalABC.NET |
| --- |
| **Var**  D, M: Integer;  **begin**  Write('Введите День: ');  Readln(D);  Write('Введите Месяц: ');  Readln(M);    **case** M \* 100 + D **of**  0120..0218: Writeln('Водолей');  0219..0320: Writeln('Рыбы');  0321..0419: Writeln('Овен');  0420..0520: Writeln('Телец');  0521..0621: Writeln('Близнецы');  0622..0722: Writeln('Рак');  0723..0822: Writeln('Лев');  0823..0922: Writeln('Дева');  0923..1022: Writeln('Весы');  1023..1122: Writeln('Скорпион');  1123..1221: Writeln('Стрелец');  1222..1231, 0101..0119: Writeln('Козерог');  **end**;  **end**. |
| C++ |
| #include <iostream>  #include <windows.h>  using namespace std;  int main() {  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  int d, m;  cout << "день:";  cin >> d;  cout << "месяц:";  cin >> m;  switch (m) {  case 1:  if (d > 19) cout << "Водолей";  else cout << "Козерог";  break;  case 2:  if (d > 18) cout << "Рыбы";  else cout << "Водолей";  break;  case 3:  if (d > 20) cout << "Овен";  else cout << "Рыбы";  break;  case 4:  if (d > 19) cout << "Телец";  else cout << "Овен";  break;  case 5:  if (d > 20) cout << "Близнецы";  else cout << "Телец";  break;  case 6:  if (d > 21) cout << "Рак";  else cout << "Близнецы";  break;  case 7:  if (d > 22) cout << "Лев";  else cout << "Рак";  break;  case 8:  if (d > 22) cout << "Дева";  else cout << "Лев";  break;  case 9:  if (d > 22) cout << "Весы";  else cout << "Дева";  break;  case 10:  if (d > 22) cout << "Скорпион";  else cout << "Весы";  break;  case 11:  if (d > 22) cout << "Стрелец";  else cout << "Скорпион";  break;  case 12:  if (d > 21) cout << "Козерог";  else cout << "Стрелец";  break;  }  cout << endl;  system("pause");  return 0;  } |

**Задачи для самостоятельной работы**

**Вариант 1**

1. Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Даны три действительные числа. Возвести в квадрат те из них, значения которых неотрицательны, и в четвертую степень – отрицательные.

3) Найти max {min (a, b), min (c, d)}.

4) Даны два различных вещественных числа. Определить:

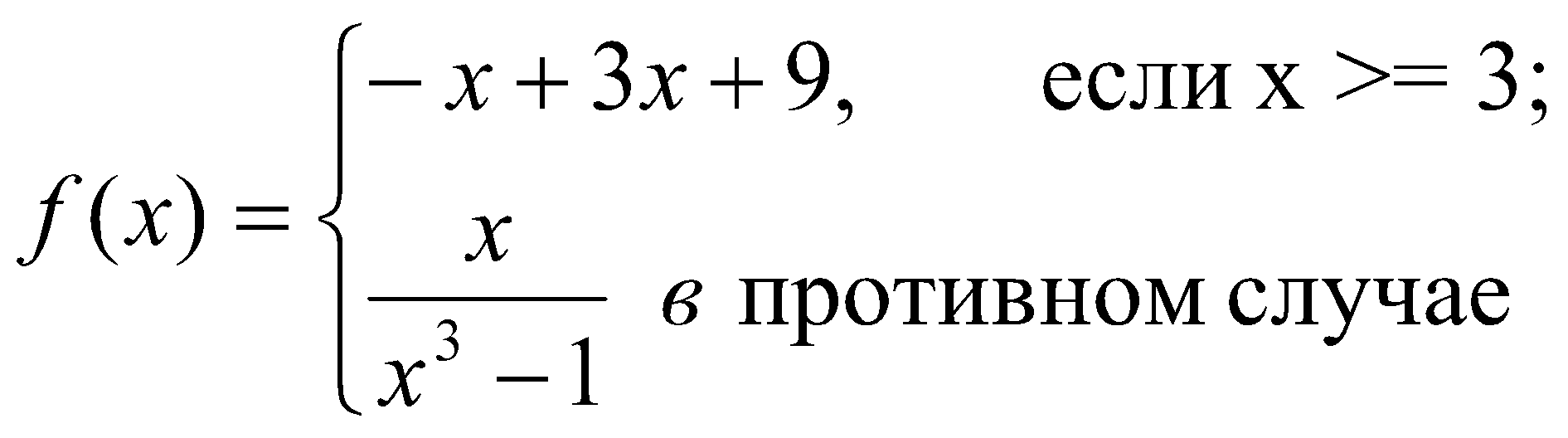
а) какое из них больше;

б) какое из них меньше.

5) Составить программу, которая в зависимости от порядкового номера дня недели (1, 2, ..., 7) выводит на экран его название (понедельник, вторник, ...,воскресенье).

**Вариант 2**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Даны две точки А(x1 , y1), В(x 2, y2). Составить алгоритм, определяющий, которая из точек находится ближе к началу координат.

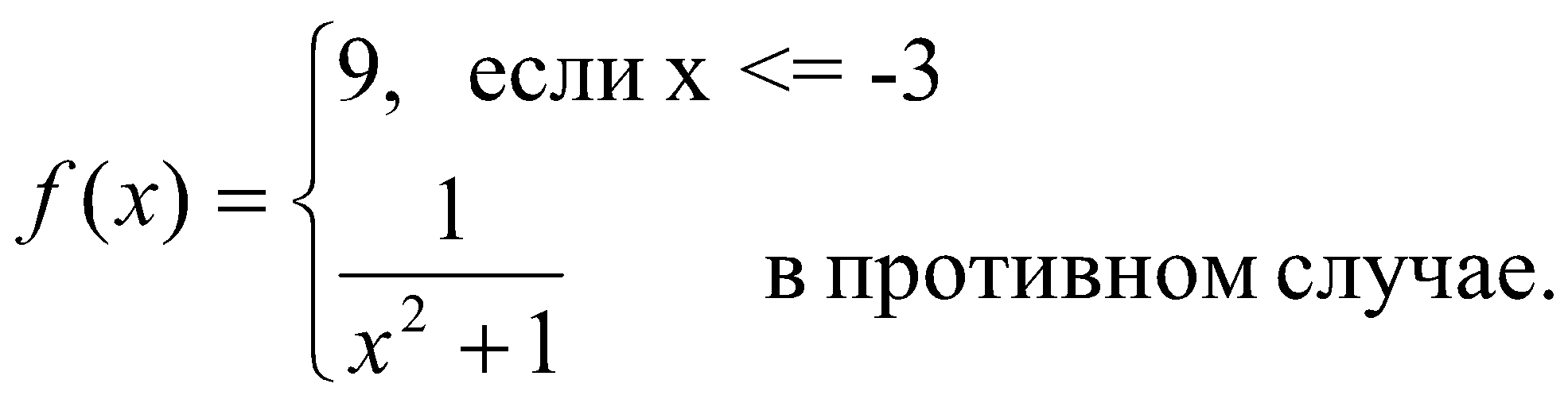
3) Даны три числа m, n, p. Определить какое из них равно d. Если ни одно не рано d, то найти max(d-m, d-n, d-p).

4) Определить максимальное и минимальное значения из двух различных вещественных чисел.

5) Мастям игральных карт условно присвоены следующие порядковые номера: масти "пики" *—* 1, масти "трефы" *—* 2, масти "бубны" *—* 3, масти "червы" *—*4. По заданному номеру масти *m* (1 *m* 4) определить название соответствующей масти.

**Вариант 3**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Даны два угла треугольника (в градусах ). Определить существует ли такой треугольник. Если да, то будет ли он прямоугольным.

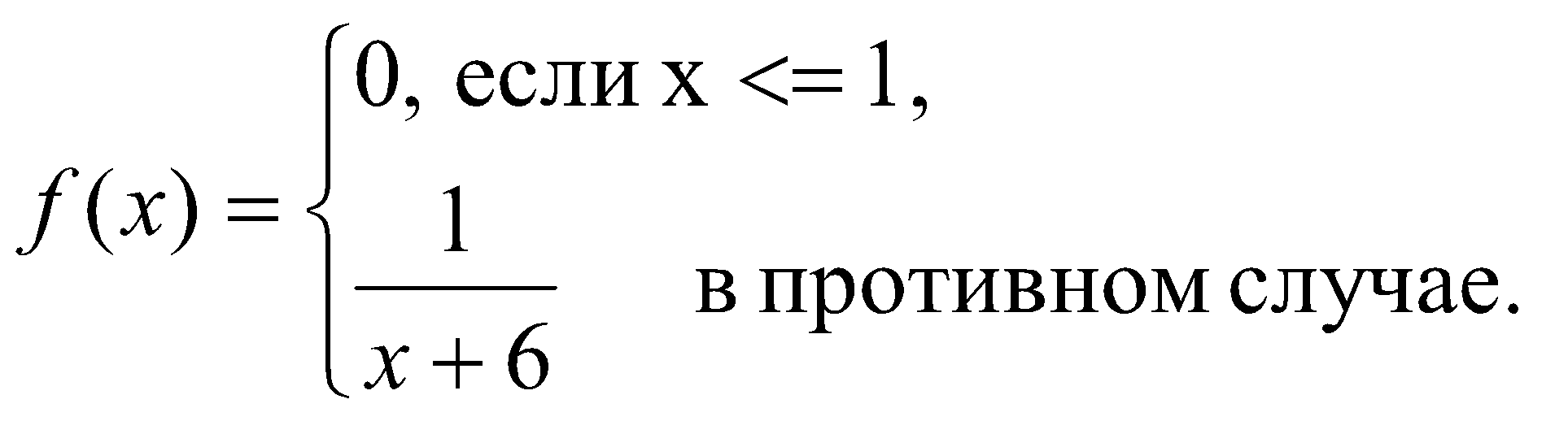
3) Даны четыре точки А1(x1, y1), А2(x2, y2), А3(x3, y3), А4(x4, y4). Определить будут ли они вершинами параллелограмма.

4) Известны два расстояния: одно в километрах, другое — в футах (1 фут 0,305 м). Какое из расстояний меньше?

5) Игральным картам условно присвоены следующие порядковые номера в зависимости от их достоинства: "валету" *—* 11, "даме" *—* 12, "королю" *—* 13, "тузу" *—* 14. Порядковые номера остальных карт соответствуют их названиям ("шестерка", "девятка" и т. п.). По заданному номеру карты *k* (6 *k* 14) определить достоинство соответствующей карты.

**Вариант 4**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2. Даны действительные числа x и y, не равные друг другу. Меньшее из этих чисел заменить их полусуммой, а большее – их удвоенным произведением.

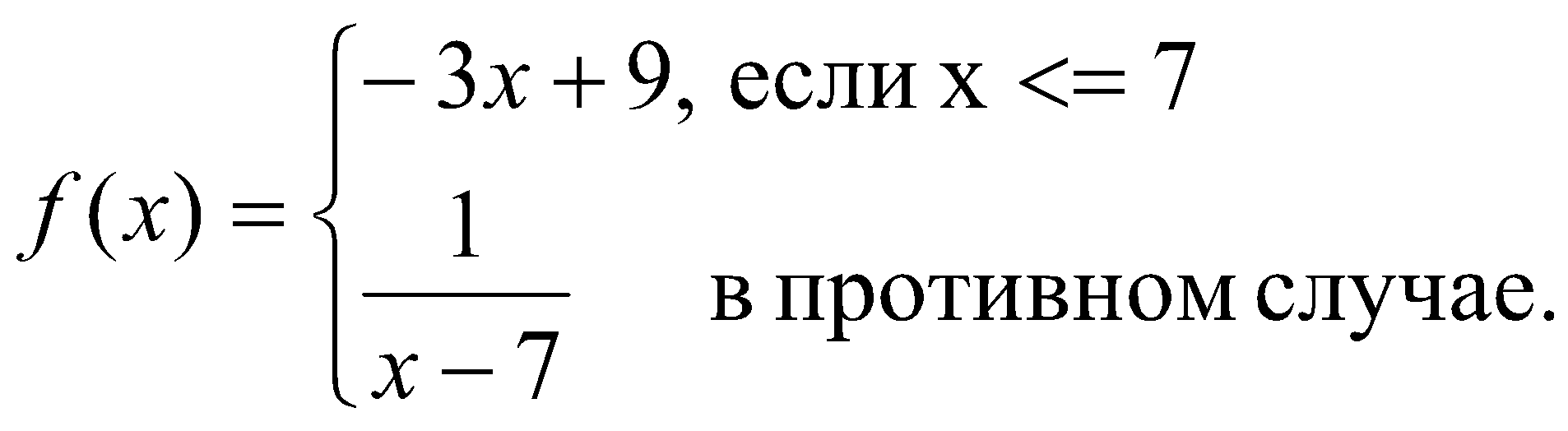
3. Даны три точки А(x1, y1), В(x2, y2), С(x3, y3). Определить будут ли они расположены на одной прямой. Если нет, то вычислить угол АВС.

4) Известны две скорости: одна в километрах в час, другая — в метрах в секунду. Какая из скоростей больше?

5) Мастям игральных карт условно присвоены следующие порядковые номера:  
масти "пики" *—* 1, масти "трефы" *—* 2, масти "бубны" *—* 3, масти "червы" *—*4, а достоинству карт: "валету" *—* 11, "даме" *—* 12, "королю" *—* 13, "тузу" *—*14 (порядковые номера карт остальных достоинств соответствуют их названиям: "шестерка", "девятка" и т. п.). По заданным номеру масти *m* (1 *m* 4) и номеру достоинства карты *k* (6 *k* 14) определить полное название (масть и достоинство) соответствующей карты в виде "Дама пик", "Шестерка

**Вариант 5**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) На плоскости XOY задана своими координатами точка А. Указать, где она расположена: на какой оси или в каком координатном угле.

3)Даны действительные числа a, b, c. Удвоить эти числа, если a>=b>=c и заменить их абсолютными значениями, если это не так.

4) Даны радиус круга и сторона квадрата. У какой фигуры площадь больше?

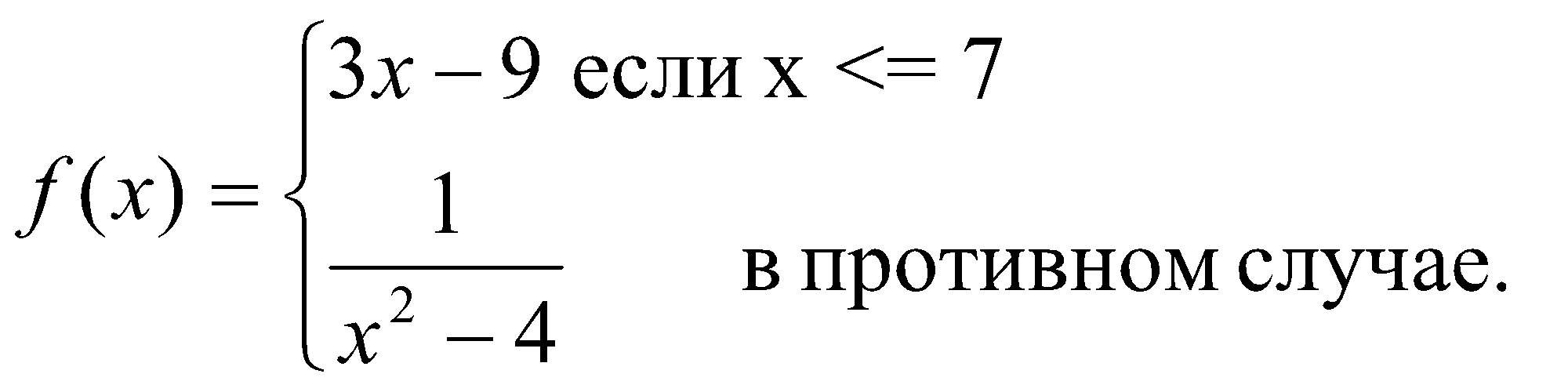
5) Дано целое число *k* (1 *k* 365). Определить, каким днем недели (понедельником, вторником, ..., субботой или воскресеньем) является *k*-й день невисокосного года, в котором 1 января:

а) понедельник;

б)*d*-й день недели (если 1 января — понедельник, то *d =*1 , если вторник — *d=* 2 , ..., если воскресенье — *d =*7).

**Вариант 6**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Даны целые числа m и n. Если числа не равны, то заменить каждое из них одним и тем же числом, равным большему из исходных, а если равны, то заменить числа нулями.

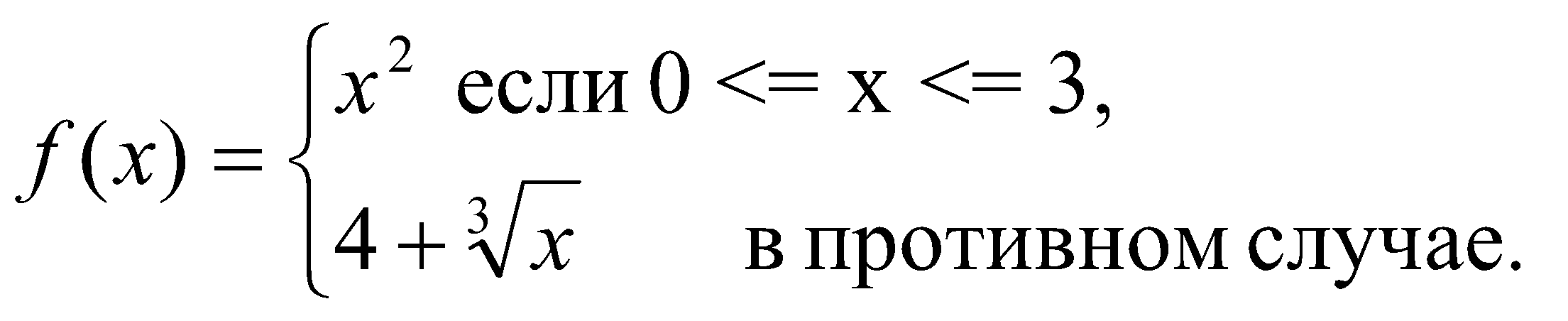
3) На оси ОХ расположены три точки a, b, c. Определить какая из точек b, c расположена ближе к а.

4) Даны объемы и массы двух тел из разных материалов. Материал какого из тел имеет большую плотность?

5) С начала 1990 года по некоторый день прошло *n* месяцев и 2 дня. Определить название месяца (январь, февраль и т. п.) этого дня.

**Вариант 7**

1. Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

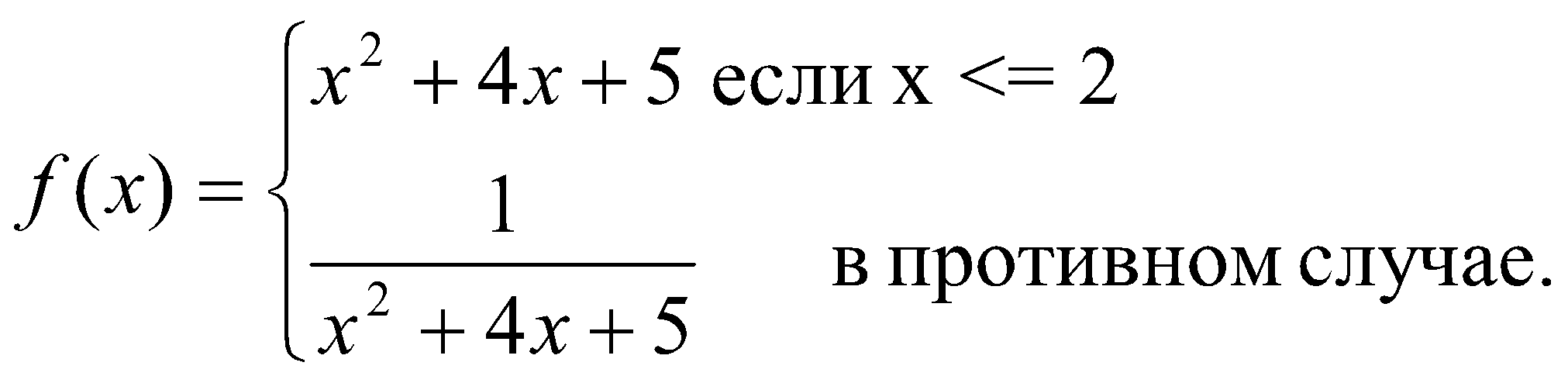


1. Дано трехзначное число N. Проверить, будет ли сумма его цифр четным числом.
2. Даны три положительных числа a, b, c. Проверить будут ли они сторонами треугольника? Если да, то вычислить площадь этого треугольника.
3. Известны сопротивления двух несоединенных друг с другом участков электрической цепи и напряжение на каждом из них. По какому участку протекает меньший ток?
4. Дата некоторого дня характеризуется двумя натуральными числами: *m* (порядковый номер месяца) и *n* (число). По заданным *n* и *m* определить:

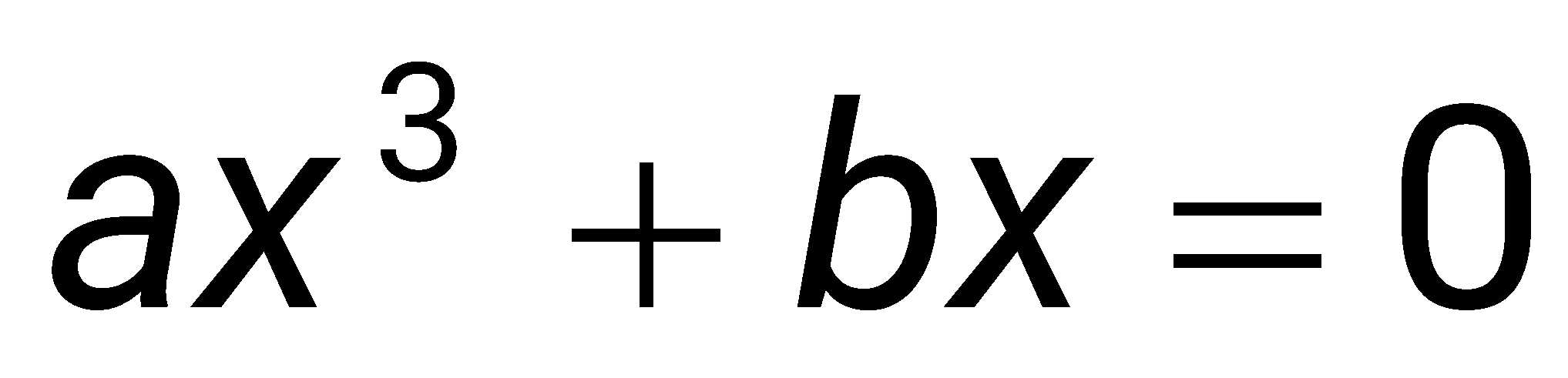
а) дату предыдущего дня (принять, что *n* и *m* не характеризуют 1 января);  
б) дату следующего дня (принять, что *n* и *m* не характеризуют 31 декабря)

**Вариант 8**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Написать программу определения принадлежности точки с координатами (x,y) части плоскости, лежащей между прямыми x=m, x=n, (m<n).

3) Написать программу решения уравнения  для произвольных a, b.

4) Даны вещественные числа *a*, *b*, *c* (). Выяснить, имеет ли уравнение вещественные корни.

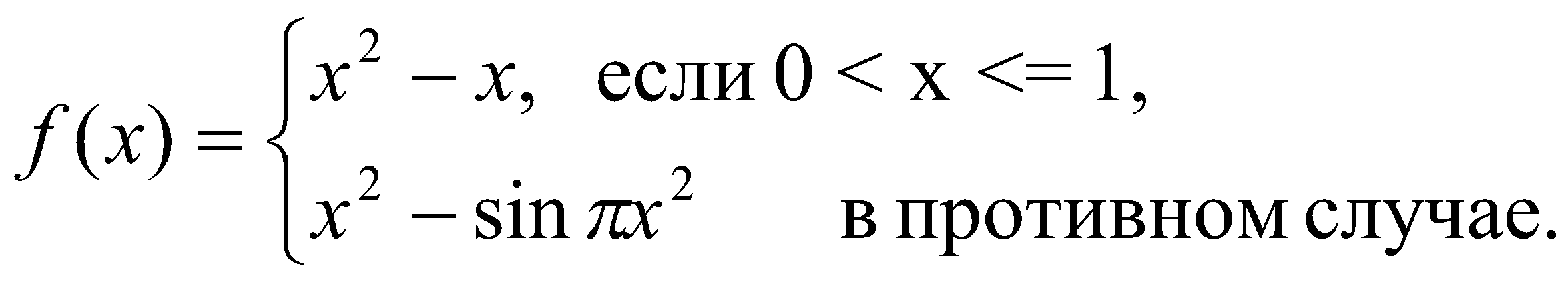
5) Дата некоторого дня характеризуется тремя натуральными числами: *g* (год), *m* (порядковый номер месяца) и *n* (число). По заданным *g*, *n* и *m* определить:

а) дату предыдущего дня;  
б) дату следующего дня.

В обеих задачах рассмотреть два случая:  
1) заданный год не является високосным;  
2) заданный год может быть високосным

**Вариант 9**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Определить, равен ли квадрат заданного трехзначного числа кубу суммы цифр этого числа.

3) Дан радиус круга R. Определить поместится ли правильный треугольник со стороной а в этом круге.

4) Известны год и номер месяца рождения человека, а также год и номер месяца сегодняшнего дня (январь — 1 и т. д.). Определить возраст человека (число полных лет). В случае совпадения указанных номеров месяцев читать, что прошел полный год.

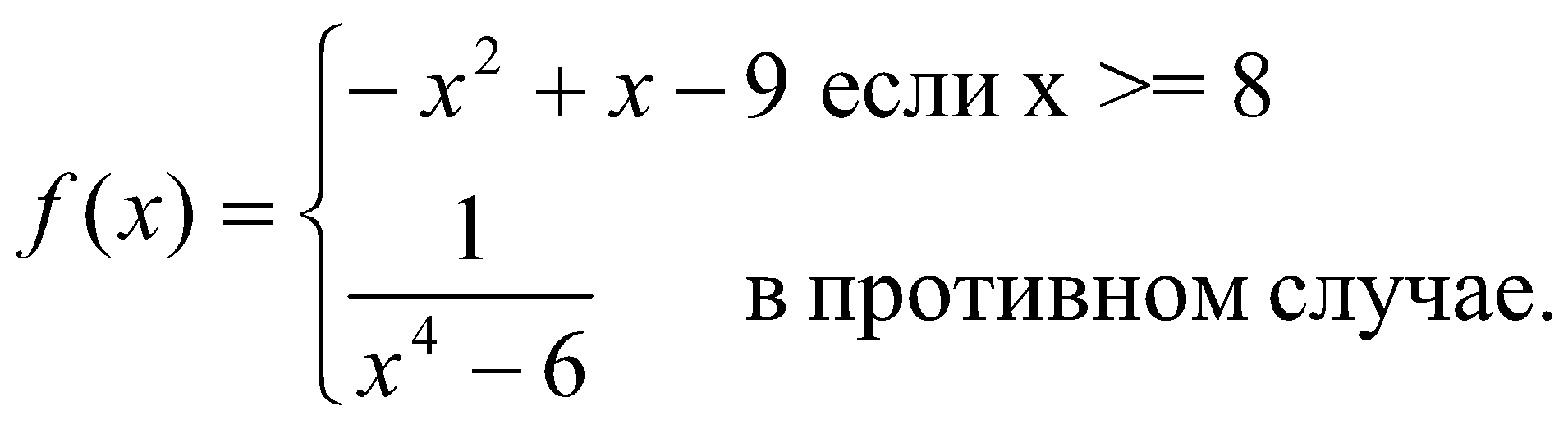
5) В некоторых странах Дальнего Востока (Китае, Японии и др.) использовался (и неофициально используется в настоящее время) календарь, отличающийся от применяемого нами. Этот календарь представляет собой 60-летнюю циклическую систему. Каждый 60-летний цикл состоит из пяти 12-летних подциклов. В каждом подцикле года носят названия животных: Крыса, Корова, Тигр, Заяц, Дракон, Змея, Лошадь, Овца, Обезьяна, Петух, Собака и Свинья. Кроме того, в названии года фигурируют цвета животных, которые связанышенно не совпадает с его "естественной" окраской — Тигр может быть черным, Свинья — красной, а Лошадь — зеленой. Например, 1984 год *—* год начала очередного цикла *—* назывался *годом Зеленой Крысы*. Каждый цвет в цикле (начиная с зеленого) "действует" два года, поэтому через каждые 60 лет имя года (животное и его цвет) повторяется.  
Составить программу, которая по заданному номеру года нашей эры *n* печатает его название по описанному календарю в виде: "Крыса, Зеленый". Рассмотреть два случая:

а) значение *n* 1984;

б) значение *n*

**Вариант 10**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Определить , является ли целое число N четным двузначным числом.

3)Дано число х. Напечатать в порядке возрастания числа sin x, cos x, ln x. Если при каком-либо х некоторые из выражений не имеют смысла, вывести сообщение об этом и сравнивать значения только тех, которые имеют смысл.

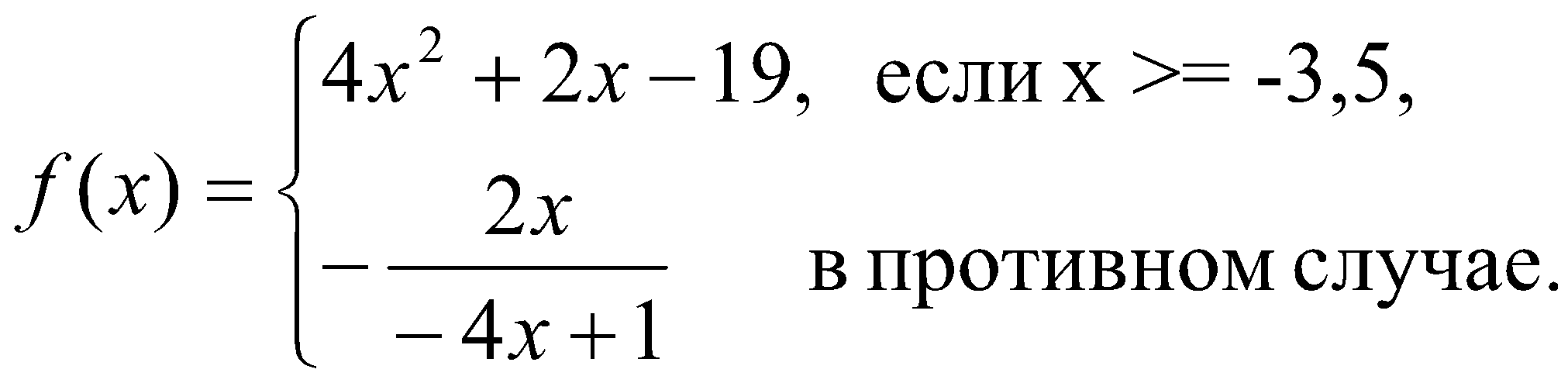
4) Известны площади круга и квадрата. Определить:

а) уместится ли круг в квадрате?  
б) уместится ли квадрат в круге?

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит 8: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, каждое из которых не превосходит 8.  
а) на поле (*a*, *b*) расположена ладья. Определить, угрожает ли она полю  
(*c*, *d*);  
б) на поле (*a*, *b*) расположен слон. Определить, угрожает ли он полю (*c*, *d*);

**Вариант 11**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Определить , является ли треугольник со сторонами a, b, c равносторонним.

3) Заданы размеры А, В прямоугольного отверстия и размеры X, Y, Z кирпича. Определить пройдет ли кирпич через отверстие.

4) Известны площади круга и равностороннего треугольника. Определить:

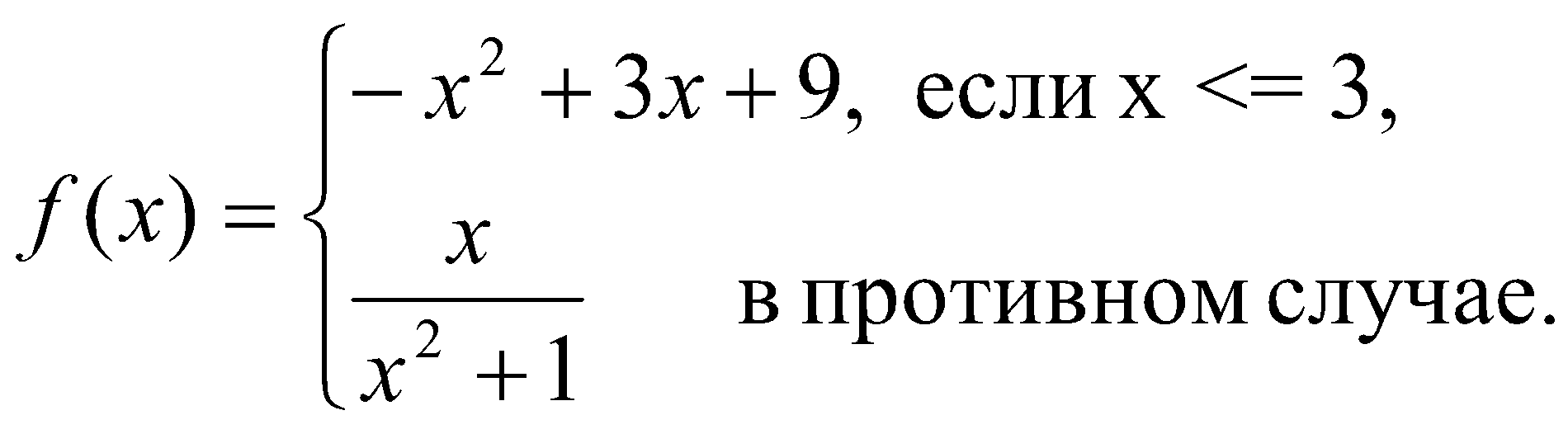
а) уместится ли круг в треугольнике?  
б) уместится ли треугольник в круге?

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит 8: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, каждое из которых не превосходит 8.

a) на поле (*a*, *b*) расположен король. Определить, может ли он одним ходом  
попасть на поле (*c*, *d*);  
б) на поле (*a*, *b*) расположен ферзь. Определить, угрожает ли он полю (*c*, *d*);

**Вариант 12**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Определить , является ли треугольник со сторонами a, b, c равнобедренным.

3) Составить программу, осуществляющую перевод величин из радианной меры в градусную или наоборот. Программа должна запрашивать, какой перевод нужно осуществить, и выполнять указанное действие.

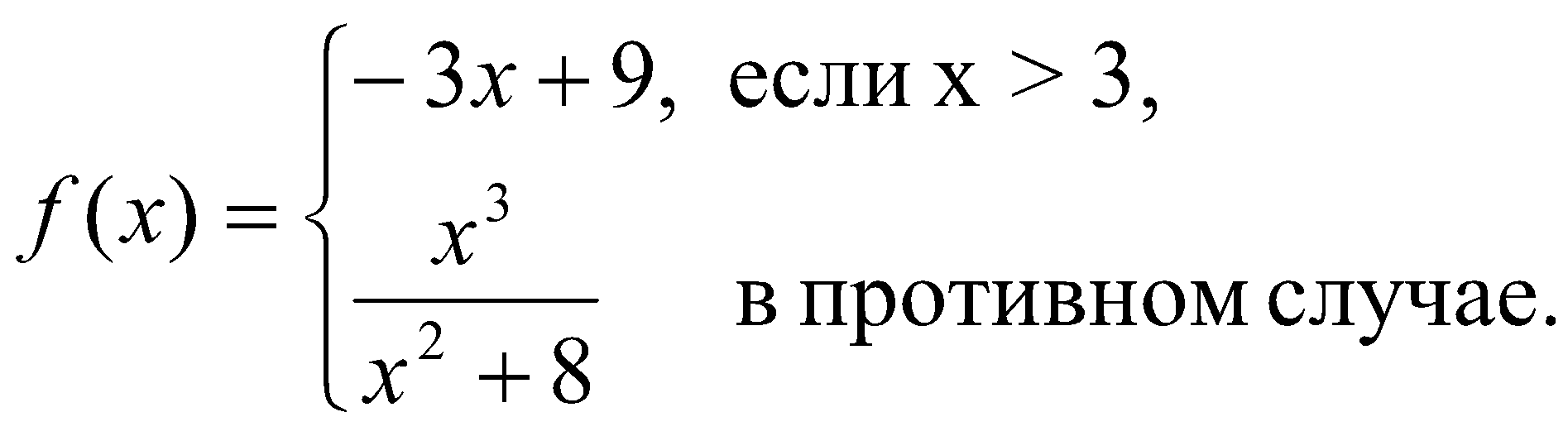
4) Даны два прямоугольника, стороны которых параллельны или перпендикулярны осям координат. Известны координаты левого нижнего и правого нижнего углов каждого из них. Найти координаты левого нижнего и правого верхнего углов минимального прямоугольника, содержащего указанные прямоугольники.

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит 8: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, каждое из которых не превосходит 8.  
На поле (*a*, *b*) расположена белая пешка. Определить, может ли она одним ходом попасть на поле (*c*, *d*):при обычном ходе; когда она "бьет" фигуру или пешку соперника;

***Примечание.*** Белые пешки перемещаются на доске снизу вверх.

**Вариант 13**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Определить, имеется ли среди чисел a, b, c хотя бы одна пара взаимно противоположных чисел.

3) Два прямоугольника, расположенные в первом квадранте, со сторонами, параллельными осям координат, заданы координатами своих левого верхнего и правого нижнего углов. Для первого прямоугольника это точки (х1, у1) и (х2, 0), для второго – (х3, у3), (х4, 0). Составить программу, определяющую, пересекаются ли данные прямоугольники, и вычисляющую площадь общей части, если они пересекаются.

4) Даны два прямоугольника, стороны которых параллельны или перпендикулярны осям координат. Известны координаты левого нижнего угла каждого из них и длины их сторон. Найти координаты левого нижнего и правого верхнего углов минимального прямоугольника, содержащего указанные прямоугольники.

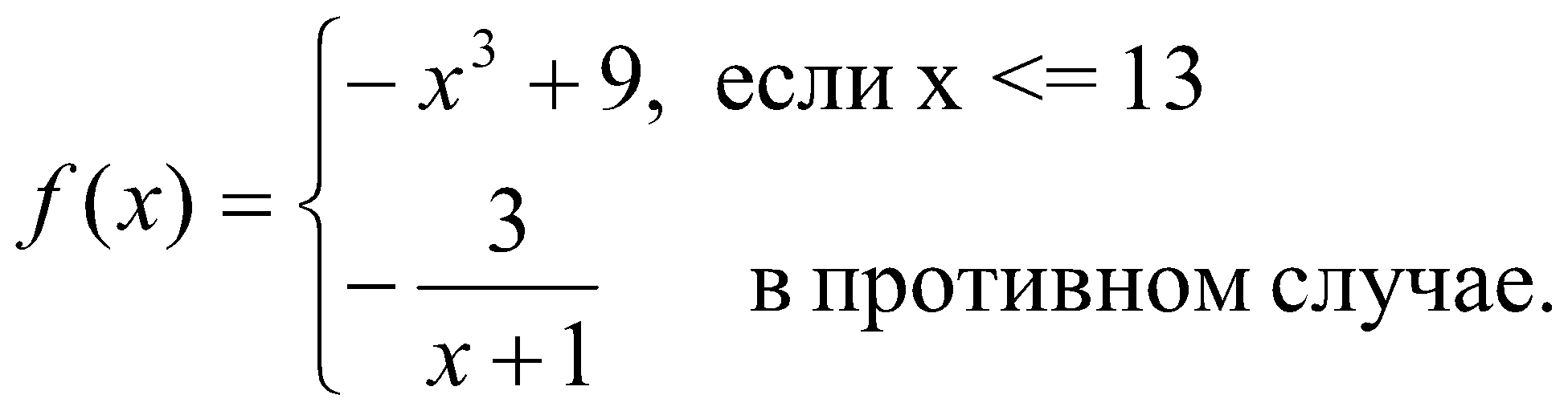
5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит 8: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, каждое из которых не превосходит 8.

На поле (*a*, *b*) расположена черная пешка. Определить, может ли она одним ходом попасть на поле (*c*, *d*): при обычном ходе; когда она "бьет" фигуру или пешку соперника;

***Примечание.*** Черные пешки перемещаются на доске сверху вниз.

**Вариант 14**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Подсчитать количество отрицательных чисел среди чисел m, n, p.

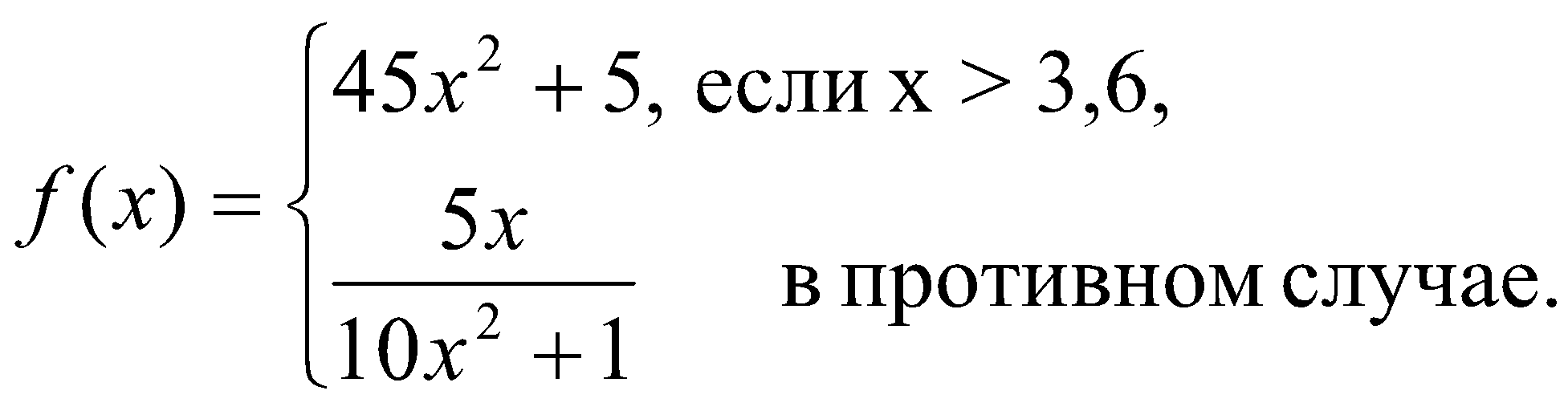
3) В небоскребе N этажей и всего один подъезд; на каждом этаже по три квартиры; лифт может останавливаться только на нечетных этажах. Человек садится в лифт и набирает номер нужной ему квартиры М. На какой этаж должен доставить лифт пассажира?

4) Если целое число *m* делится нацело на целое число *n*, то вывести на экран частное от деления, в противном случае вывести сообщение "*m* на *n* нацело не  
делится".

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит 8: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, каждое из которых не превосходит 8.  
а) на поле (*a*, *b*) расположен конь. Определить, угрожает ли он полю (*c*, *d*).

**Вариант 15**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Подсчитать количество положительных чисел среди чисел m, n, p.

3) Написать программу, которая по заданным трем числам определяет, является ли сумма каких-либо двух из них положительной.

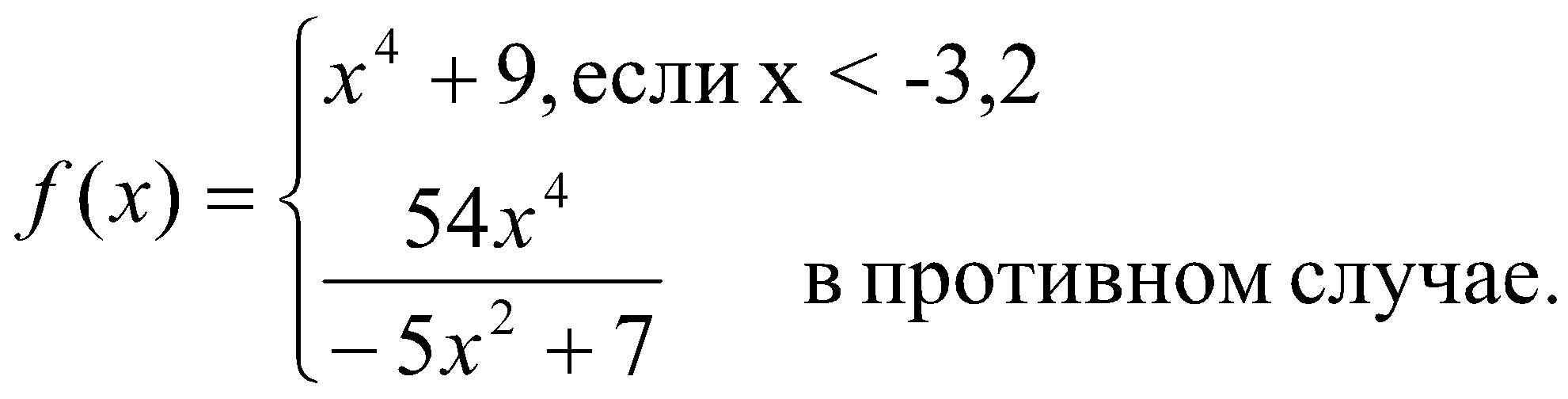
4) Определить, является ли число *a* делителем числа *b*?

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, *e*, *f*, каждое из которых не превосходит восьми. На поле (*a*, *b*) расположена белая фигура, на поле (*c*, *d*) — черная. Определить, может ли белая фигура пойти на поле (*e*, *f*), не попав при этом под удар черной фигуры.

Рассмотреть следующие варианты сочетаний белой и черной фигур:

а) ладья и ладья;  
б) ладья и ферзь;  
в) ладья и конь;  
**Вариант 16**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2. Числа a и b выражают длины катетов одного прямоугольного треугольника, а c и d – другого. Определить, являются ли эти треугольники подобными.

3. Известно, что из четырех чисел а1, а2, а3, а4 одно отлично от трех других, равных между собой; присвоить номер этого числа переменной n.

4) Дано натуральное число. Определить:

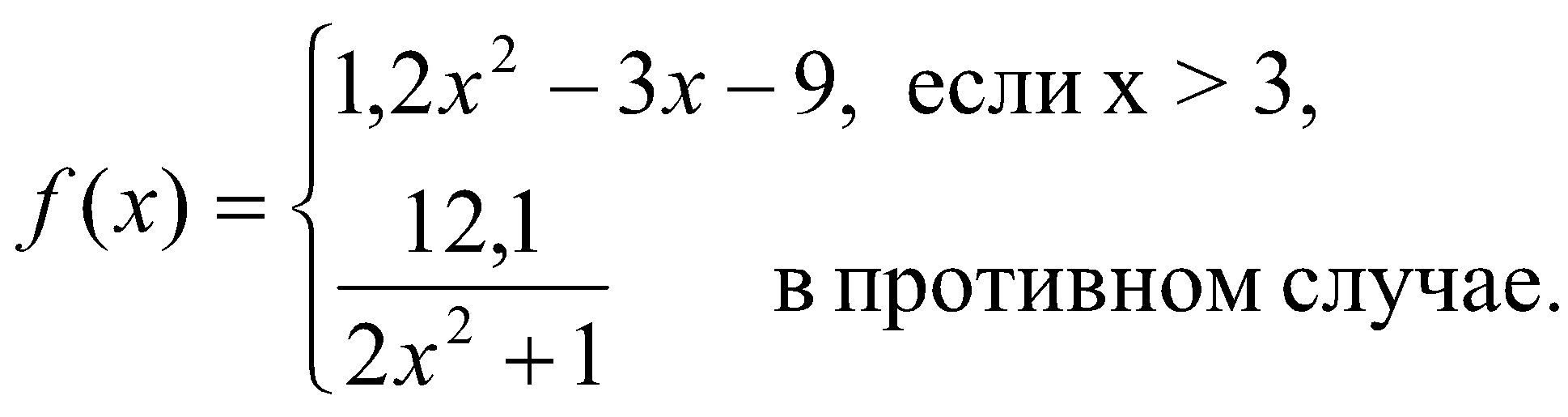
а) является ли оно четным;  
б) оканчивается ли оно цифрой 7.

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, *e*, *f*, каждое из которых не превосходит восьми. На поле (*a*, *b*) расположена белая фигура, на поле (*c*, *d*) — черная. Определить, может ли белая фигура пойти на поле (*e*, *f*), не попав при этом под удар черной фигуры.  
Рассмотреть следующие варианты сочетаний белой и черной фигур:

а) ладья и слон;  
б) ферзь и ферзь;  
в) ферзь и ладья;

**Вариант 17**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Подсчитать количество целых чисел среди чисел m, n, p.

3) Составить программу, которая проверяла бы, не приводит ли суммирование двух целых чисел А и В к переполнению (т.е. к результату большему, чем 32767). Если будет переполнение, то сообщить об этом, иначе вывести сумму этих чисел.

4) Дано двузначное число. Определить:

а) какая из его цифр больше: первая или вторая;

б) одинаковы ли его цифры.

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, *e*, *f*, каждое из которых не превосходит восьми. На поле (*a*, *b*) расположена белая фигура, на поле (*c*, *d*) — черная. Определить, может ли белая фигура пойти на поле (*e*, *f*), не попав при этом под удар черной фигуры.

Рассмотреть следующие варианты сочетаний белой и черной фигур:

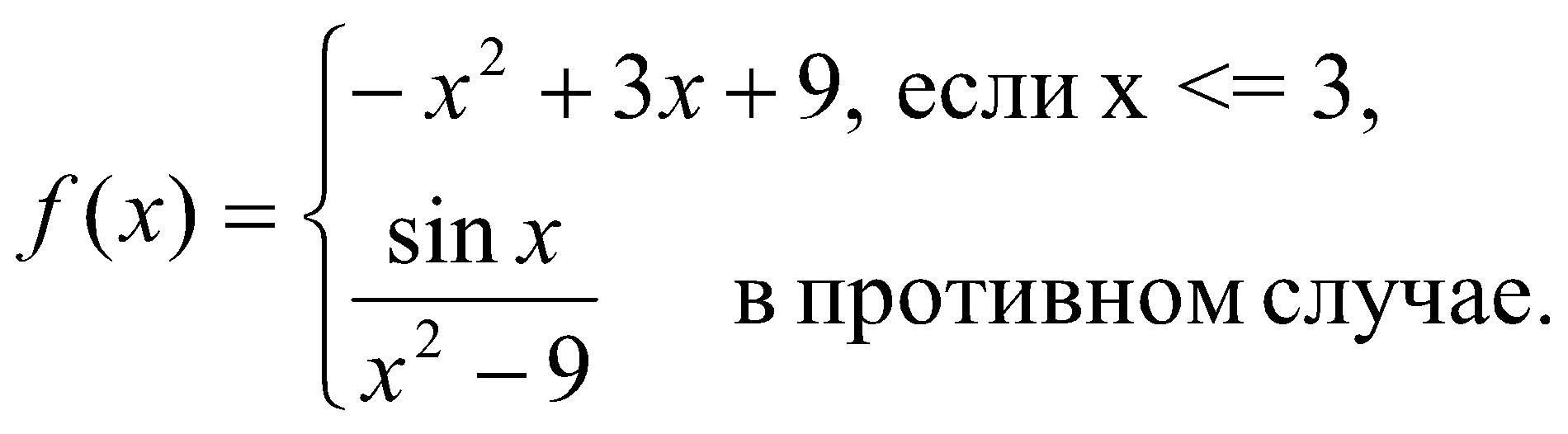
а) ферзь и конь;

б) ферзь и слон;

в) конь и конь;

**Вариант 18**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Определить, делителем каких чисел m, n, p является число a.

3) Перераспределить значения переменных х и у так, чтобы в х оказалось большее из этих значений, а в у— меньшее.

4) Дано двузначное число. Определить, равен ли квадрат этого числа учетверенной сумме кубов его цифр. Например, для числа 48 ответ положительный, для числа 52 — отрицательный.

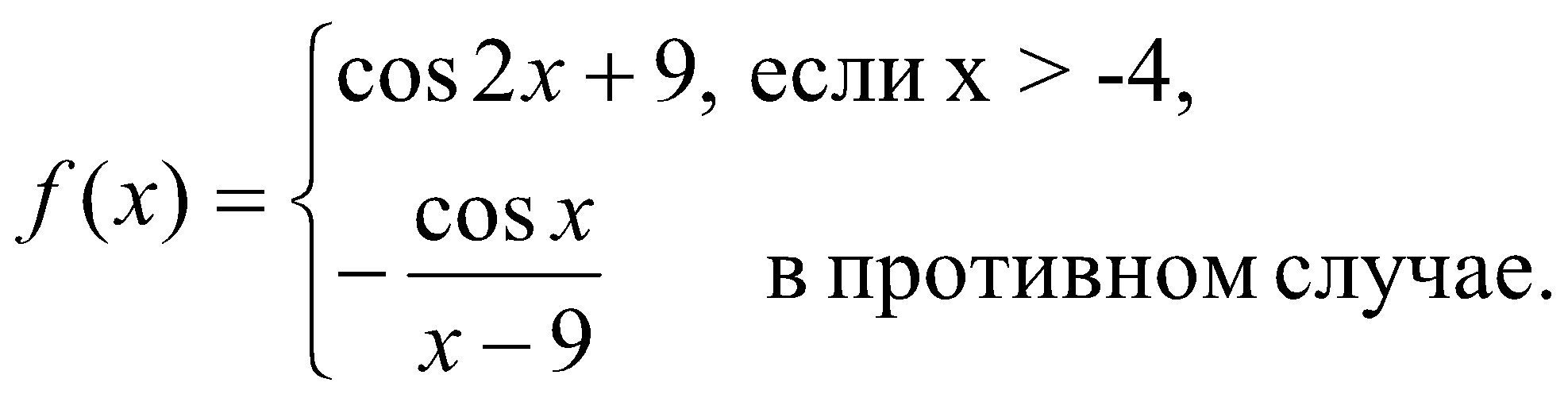
5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, *e*, *f*, каждое из которых не превосходит восьми. На поле (*a*, *b*) расположена белая фигура, на поле (*c*, *d*) — черная. Определить, может ли белая фигура пойти на поле (*e*, *f*), не попав при этом под удар черной фигуры.

Рассмотреть следующие варианты сочетаний белой и черной фигур:

а) конь и ладья;  
б) конь и ферзь;  
в) конь и слон;

**Вариант 19**

1)Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Услуги телефонной сети оплачиваются по следующему правилу: за разговоры за А минут в месяц оплачиваются В р., а разговоры сверх установленной нормы оплачиваются из расчета С р. в минуту. Написать программу, вычисляющую плату за пользование телефоном для введенного времени разговоров за месяц.

3) Известно, что из четырех чисел а1, а2, а3 и а4 одно отлично от трех других, равных между собой; присвоить номер этого числа переменной n.

4) Дано двузначное число. Определить:

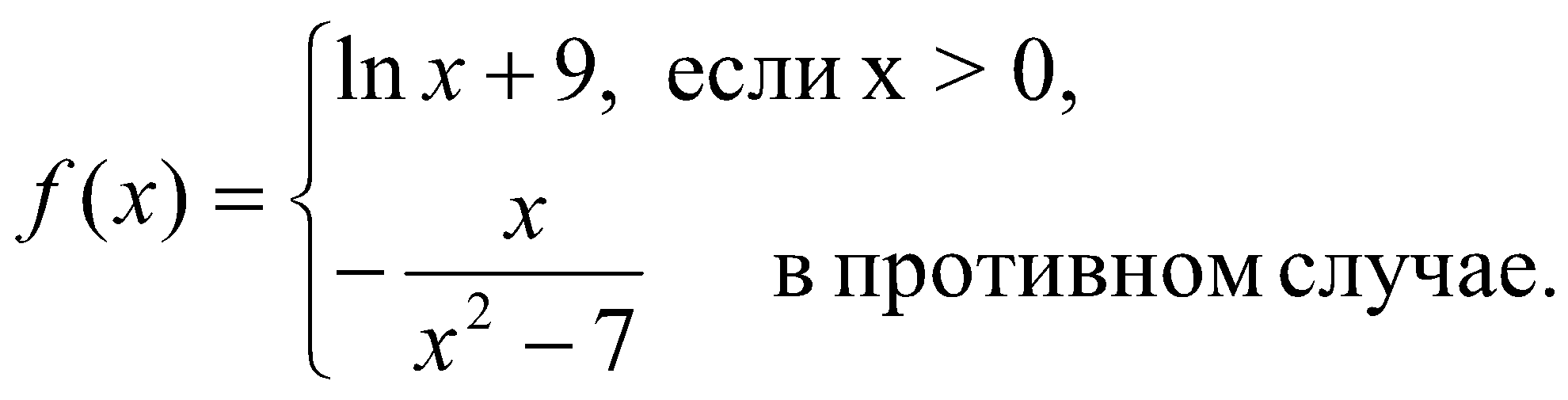
а) является ли сумма его цифр двузначным числом;  
б) больше ли числа *а* сумма его цифр.

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, *e*, *f*, каждое из которых не превосходит восьми. На поле (*a*, *b*) расположена белая фигура, на поле (*c*, *d*) — черная. Определить, может ли белая фигура пойти на поле (*e*, *f*), не попав при этом под удар черной фигуры.  
Рассмотреть следующие варианты сочетаний белой и черной фигур:

а) слон и слон;  
б) слон и ферзь;  
в) слон и конь;

**Вариант 20**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Даны три стороны одного и три стороны другого треугольника. Определить, будут ли эти треугольники равновеликими, т.е. имеют ли они равные площади.

3) Значения переменных а, Ь и с поменять местами так, чтобы оказалось а ≥ Ь ≥ с.

4) Дано двузначное число. Определить:

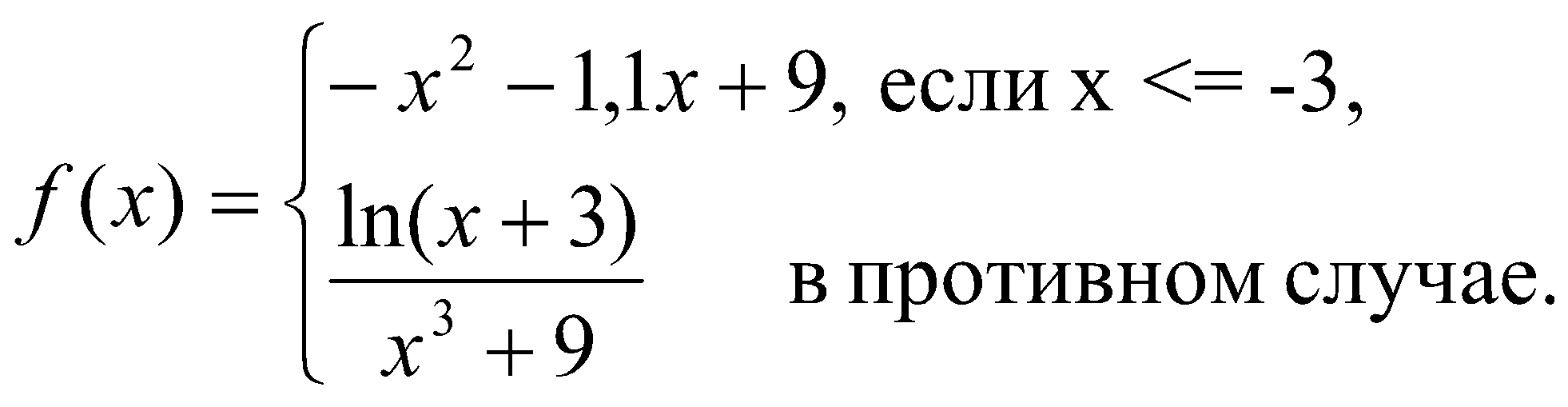
а) кратна ли трем сумма его цифр;  
б) кратна ли сумма его цифр числу *а*.

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, *e*, *f*, каждое из которых не превосходит восьми. На поле (*a*, *b*) расположена белая фигура, на поле (*c*, *d*) — черная. Определить, может ли белая фигура пойти на поле (*e*, *f*), не попав при этом под удар черной фигуры.  
Рассмотреть следующие варианты сочетаний белой и черной фигур:

а) слон и ладья;  
б) король и слон;  
в) король и ферзь;

**Вариант 21**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Программа-льстец. На экране появляется вопрос «Кто ты: мальчик или девочка?». В зависимости от ответа на экране должен появиться текст «Мне нравятся мальчики!» или «Мне нравятся девочки!».

3) Если уравнение ах2 + bх + с = 0 (а≠0) имеет вещественные корни, то логической переменной t присвоить значение true, а переменным х1 и х2 — сами корни, иначе же переменной t присвоить false, а значения переменных х1 и х2 не менять;

4) Дано трехзначное число. Выяснить, является ли оно палиндромом ("перевертышем"), т. е. таким числом, десятичная запись которого читается одинаково слева направо и справа налево.

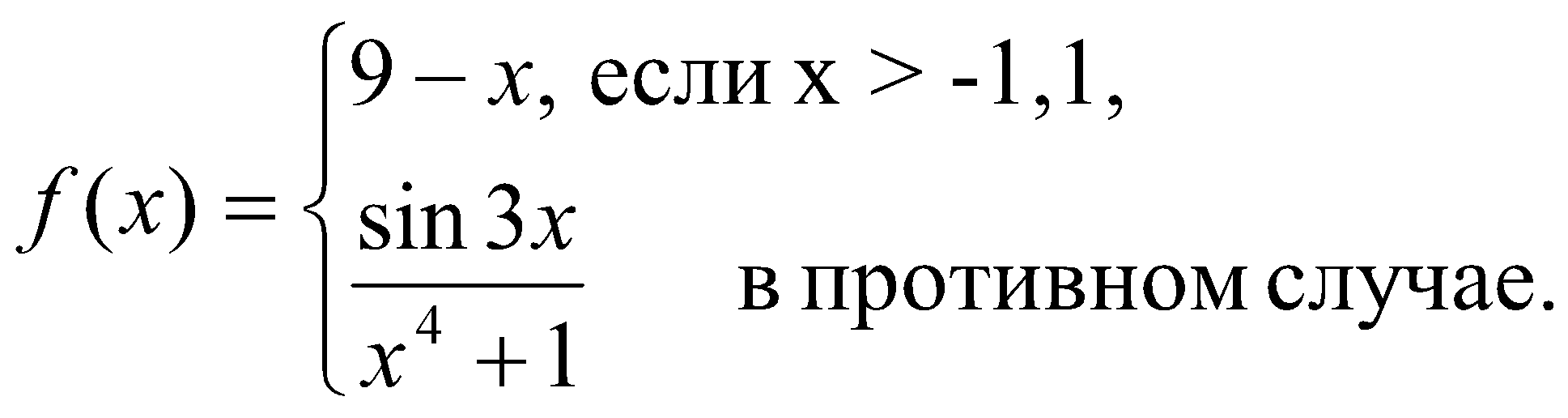
5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, *e*, *f*, каждое из которых не превосходит восьми. На поле (*a*, *b*) расположена белая фигура, на поле (*c*, *d*) — черная. Определить, может ли белая фигура пойти на поле (*e*, *f*), не попав при этом под удар черной фигуры.

Рассмотреть следующие варианты сочетаний белой и черной фигур:

а) король и конь;  
б) король и ладья.

**Вариант 22**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Грузовой автомобиль выехал из одного города в другой со скоростью v1 км/ч. Через t часов в этом же направлении выехал легковой автомобиль со скоростью v2 км/ч. Составить ппрограмму, определяющую, догонит ли легковой автомобиль грузовой через t1 ч после своего выезда.

3) Считая, что стандартные функции sin и cos применимы только к аргументам из отрезка [0, π/2], вычислить y=sin х для произвольного числа х;

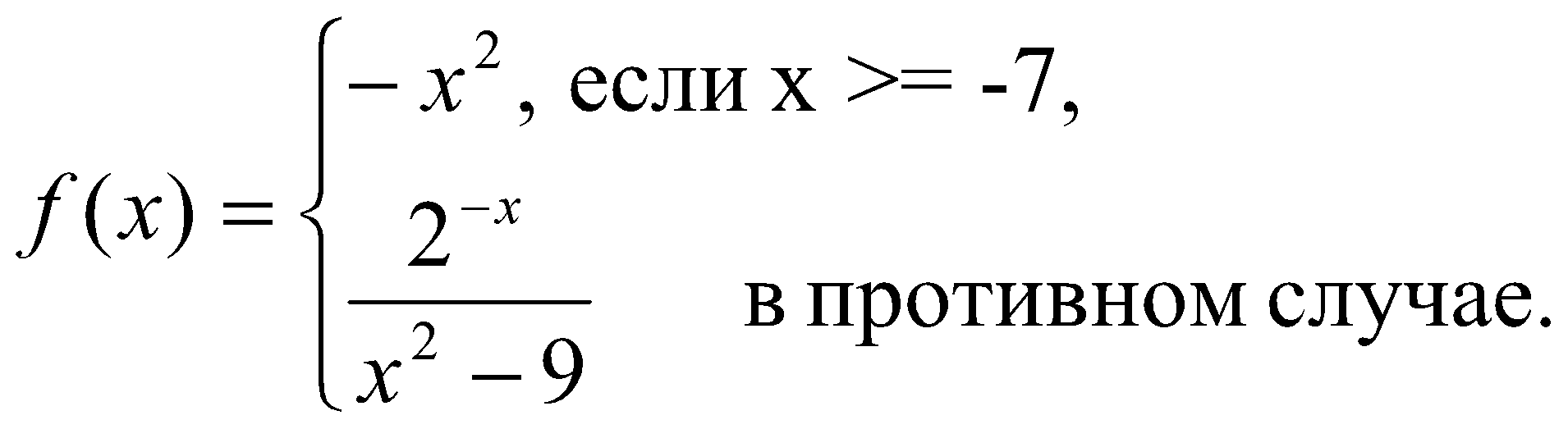
4) Дано трехзначное число. Определить, какая из его цифр больше:

а) первая или последняя;  
б) первая или вторая;  
в) вторая или последняя

5) Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми: первое число — номер вертикали (при счете слева направо), второе — номер горизонтали (при счете снизу вверх). Даны натуральные числа *a*, *b*, *c*, *d*, каждое из которых не превосходит восьми. Определить, являются ли поля (*a*, *b*) и (*c*, *d*) полями одного цвета.

**Вариант 23**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Перераспределить значения переменных x и y так, чтобы в x оказалось большее из этих значений, а в y – меньшее.

3) Если значение переменной w не равно 0 и при этом котангенс от w меньше 0.5, тогда поменять знак у w, а если значение w равно 0, тогда присвоить w значение 1.

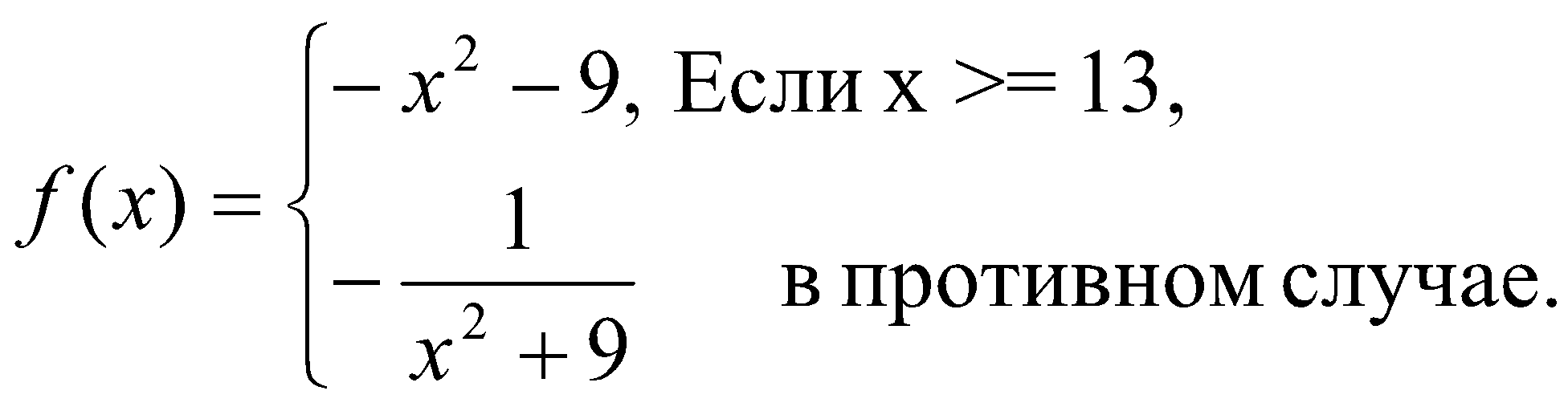
4) Дано трехзначное число. Определить, равен ли квадрат этого числа сумме кубов его цифр.

5) Даны вещественные положительные числа *a*, *b*, *c*. Если существует треугольник со сторонами *a*, *b*, *c*, то:

а) определить его вид (прямоугольный, остроугольный или тупоугольный);  
б) определить его вид (прямоугольный, остроугольный или тупоугольный) и особенности (равносторонний, равнобедренный, разносторонний).

**Вариант 24**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Определить правильность даты, введенной с клавиатуры (число – от 1 до 31, месяц – от 1 до 12). Если введены некорректные данные, то сообщить об этом.

3) Логической переменной Ь присвоить значение true, если числа х и у равны, и значение false иначе.

4) Дано трехзначное число. Определить:

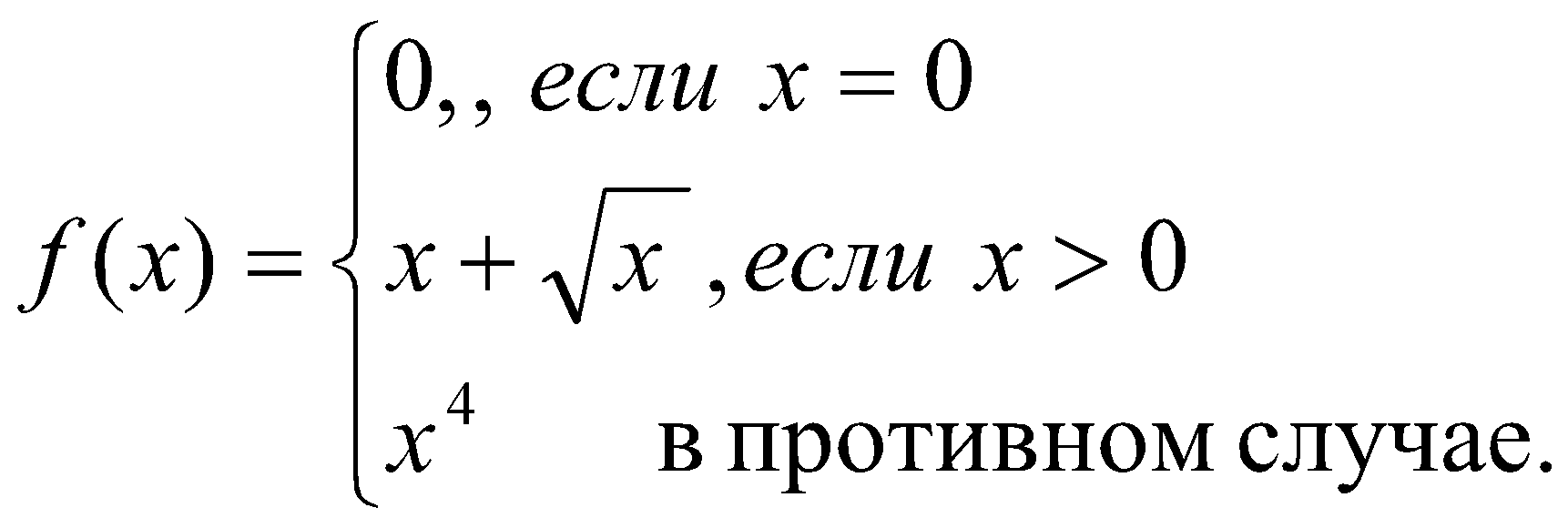
а) является ли сумма его цифр двузначным числом;  
б) является ли произведение его цифр трехзначным числом;

в) больше ли числа *а* произведение его цифр;  
г) кратна ли пяти сумма его цифр;  
д) кратна ли сумма его цифр числу *а*.

5) Дано целое число *n* (1 *n* 99), определяющее возраст человека (в годах). Для этого числа напечатать фразу "мне *n* лет", учитывая, что при некоторых значениях *n* слово "лет" надо заменить на слово "год" или "года".

**Вариант 25**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:



2) Составить программу, определяющую результат гадания по ромашке – “любит – не любит”, взяв за исходное данное количество лепестков n.

3) Даны числа a1, b1, c1, a2, b2, c2. Напечатать координаты точки пересечения прямых, описываемых уравнениями a1x+b1y=c1 и a2x+b2y=c2, либо сообщить, что эти прямые совпадают, не пересекаются или вовсе не существуют.

4) Дано трехзначное число.

а) Верно ли, что все его цифры одинаковые?  
б) Определить, есть ли среди его цифр одинаковые.

5) Дано натуральное число *n* (1 *n* 9999), определяющее стоимость товара в копейках. Выразить стоимость в рублях и копейках, например, 3 рубля 21 копейка, 15 рублей 5 копеек, 1 рубль ровно и т. п.

**Вариант 26**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

| *f*(*x*) = |  | 2·sin(*x*), | если *x* > 0, |
| --- | --- | --- | --- |
|  | 6 – *x*, | если *x* = 0. |

2) Даны целочисленные координаты точки на плоскости. Если точка совпадает с началом координат, то вывести 0. Если точка не совпадает с началом координат, но лежит на оси *OX* или *OY*, то вывести соответственно 1 или 2. Если точка не лежит на координатных осях, то вывести 3.

3-4) Дано четырехзначное число. Определить:

а) равна ли сумма двух первых его цифр сумме двух его последних цифр;

б) кратна ли трем сумма его цифр;  
в) кратно ли четырем произведение его цифр;  
г) кратно ли произведение его цифр числу *а*.

5) Дано натуральное число *n* (1 *n* 1188), определяющее возраст человека (в месяцах). Выразить возраст в годах и месяцах, например, 21 год 10 месяцев, 52 года 1 месяц, 46 лет ровно и т. п.

**Вариант 27**

1) Для данного целого *x* найти значение следующей функции *f*, принимающей значения целого типа:

| *f*(*x*) = |  | 2·*x*, | если *x* < –2 или *x* > 2, |
| --- | --- | --- | --- |
|  | –3·*x*, | в противном случае. |

2) Вывести на экран номер четверти координатной плоскости, которой принадлежит точка с координатами (*х*, *у*), при условии, что и *y* 0.

3) Даны целочисленные координаты трех вершин прямоугольника, стороны которого параллельны координатным осям. Найти координаты его четвертой вершины.

4) Дано натуральное число.

а) Верно ли, что оно заканчивается четной цифрой?  
б) Верно ли, что оно заканчивается нечетной цифрой?

5) Известны год, номер месяца и день рождения человека, а также год, номер месяца и день сегодняшнего дня. Определить возраст человека (число полных лет).

**Вариант 28**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Составить программу нахождения произведения двух наименьших из трех различных чисел.

3) Дан номер года (положительное целое число). Определить количество дней в этом году, учитывая, что обычный год насчитывает 365 дней, а високосный — 366 дней. Високосным считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400 (например, годы 300, 1300 и 1900 не являются високосными, а 1200 и 2000 — являются).

4) Определить, является ли число *а* делителем числа *b*? А наоборот? (Получить два ответа.)

5) Известны год и номер месяца рождения человека, а также год и номер месяца сегодняшнего дня. Определить возраст человека (число полных лет и число полных месяцев). При определении числа полных месяцев дни месяца не учитывать, а использовать разность между номерами месяцев. Например, если месяц рождения — февраль, а текущий (сегодняшний) месяц — май, то число.

**Вариант 29**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны три числа. Найти среднее из них (то есть число, расположенное между наименьшим и наибольшим).

3) Определить максимальное и минимальное значения из трех различных вещественных чисел.

4) Имеется стол прямоугольной формы с размерами (*a* и *b* — целые числа, *a* > *b*). В каком случае на столе можно разместить большее количество картонных прямоугольников с размерами (*c* и *d* — целые числа, *c* > *d*): при размещении их длинной стороной вдоль длинной стороны стола или вдоль короткой. Прямоугольники не должны лежать один на другом и не должны свисать со стола.

5) Дата некоторого дня определяется тремя натуральными числами: *g* (год), *m* (порядковый номер месяца) и *n* (число). По заданным *g*, *n* и *m* определить:

а) дату предыдущего дня;  
б) дату следующего дня.

В обеих задачах рассмотреть два случая:  
1) заданный год не является високосным;  
2) заданный год может быть високосным

**Вариант 30**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей значения целого типа:

2) Составить программу нахождения суммы двух наибольших из трех различных чисел.

3) Дано целое число, лежащее в диапазоне 1–999. Вывести его строку-описание вида «четное двузначное число», «нечетное трехзначное число» и т. д.

4) Работа светофора для пешеходов запрограммирована следующим образом: в начале каждого часа в течение трех минут горит зеленый сигнал, затем в течение двух минут *—* красный, в течение трех минут *—* опять зеленый и т. д. Дано вещественное число *t*, означающее время в минутах, прошедшее с начала очередного часа. Определить, сигнал какого цвета горит для пешеходов в этот момент.

5) Дано целое число *k* (1<=*k <=*365). Определить, каким будет *k*-й день года: субботой, воскресеньем или рабочим днем, если 1 января:

а) понедельник;  
б) *d*-й день недели (если 1 января — понедельник, то *d=* 1 , если вторник —*d =*2, ..., если воскресенье — *d=* 7).

**Вариант 31**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

|  |  |  |  |
| --- | --- | --- | --- |

2) Дано целое число. Вывести его строку-описание вида «отрицательное четное число», «нулевое число», «положительное нечетное число» и т. д.

3) Даны три различных целых числа. Определить, какое из них (первое, второе или третье):

а) самое большое;  
б) самое маленькое;  
в) является средним (*средним* назовем число, которое больше наименьшего из  
данных чисел, но меньше наибольшего).

4) Проверить, принадлежит ли число, введенное с клавиатуры, интервалу (–5, 3)?

5) Даны целое число *k* (1 *k* 180) и последовательность цифр 10111213...9899, в которой выписаны подряд все двузначные числа. Определить *k*-ю цифру.

***Примечание.*** Величины строкового типа не использовать.

**Вариант 32**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) На числовой оси расположены три точки: *A*, *B*, *C*. Определить, какая из двух последних точек (*B* или *C*) расположена ближе к *A*, и вывести эту точку и ее расстояние от точки *A*.

3) Даны целочисленные координаты точки на плоскости. Если точка совпадает с началом координат, то вывести 0. Если точка не совпадает с началом координат, но лежит на оси *OX* или *OY*, то вывести соответственно 1 или 2. Если точка не лежит на координатных осях, то вывести 3.

4) Даны три вещественных числа *a*, *b*, *c*. Проверить:

а) выполняется ли неравенство *a* < *b* < *c*;  
б) выполняется ли неравенство *b* > *a* > *c.*

5) Дана последовательность цифр, представляющая собой записанные подряд ноль и 20 первых натуральных чисел. Найти цифру с номером *n* в этой последовательности (1 <=*n<=*32).

***Примечание.*** Величины строкового типа не использовать.

**Вариант 33**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны три целых числа, одно из которых отлично от двух других, равных между собой. Определить порядковый номер числа, отличного от остальных.

3) Даны четыре целых числа, одно из которых отлично от трех других, равных между собой. Определить порядковый номер числа, отличного от остальных.

4) Определить, является ли число *а* делителем числа *b* или, наоборот, число *b* делителем числа *а*. Ответом должны служить сообщения "Да, одно из чисел является делителем другого" или "Нет, ни одно из чисел не является делителем другого".

5) Даны целое число *k* (1 *k* 252) и последовательность цифр 505152...9899100101...149150, в которой выписаны подряд все натуральные числа от 50 до 250. Определить *k*-ю цифру.

***Примечание.*** Величины строкового типа не использовать.

**Вариант 34**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны три числа. Вывести вначале наименьшее, а затем наибольшее из данных чисел.

3) Даны три числа. Найти сумму двух наибольших из них.

4) Определить, верно ли, что при делении неотрицательного целого числа *a* на положительное число *b* получается остаток, равный одному из двух заданных чисел *c* или *d*.

5) В подъезде жилого дома имеется *n* квартир, пронумерованных подряд, начиная с номера *a*. Определить, является ли сумма номеров всех квартир четным числом. Формулу суммы членов арифметической прогрессии не использовать.

**Вариант 35**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны два числа. Вывести порядковый номер меньшего из них.

3) Даны два числа. Вывести вначале большее, а затем меньшее из них.

4) Даны три вещественных числа *a*, *b*, *c*. Определить, имеется ли среди них хотя бы одна пара равных между собой чисел.

**Вариант 36**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны три целых числа. Найти количество положительных и количество отрицательных чисел в исходном наборе.

3) Даны два числа. Вывести большее из них.

4) Определить, является ли треугольник со сторонами *a*, *b*, *c* равносторонним.

**Вариант 37**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны три целых числа. Найти количество положительных чисел в исходном наборе.

3) Известен рост трех человек. Определить, одинаков ли их рост?

4) Определить, является ли треугольник со сторонами *a*, *b*, *c* равнобедренным.

5) Даны два целых числа: *D* (день) и *M* (месяц), определяющие правильную дату. Вывести знак Зодиака, соответствующий этой дате: «Водолей» (20.1–18.2), «Рыбы» (19.2–20.3), «Овен» (21.3–19.4), «Телец» (20.4–20.5), «Близнецы» (21.5–21.6), «Рак» (22.6–22.7), «Лев» (23.7–22.8), «Дева» (23.8–22.9), «Весы» (23.9–22.10), «Скорпион» (23.10–22.11), «Стрелец» (23.11–21.12), «Козерог» (22.12–19.1).

**Вариант 38**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) В чемпионате по футболу команде за выигрыш дается 3 очка, за проигрыш *—*0, за ничью *—* 1. Известно количество очков, полученных командой за игру. Определить словесный результат игры (выигрыш, проигрыш или ничья).

3) Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае не изменять его. Вывести полученное число.

4) Даны вещественные положительные числа *a*, *b*, *c*, *d*. Выяснить, можно ли прямоугольник со сторонами *a*, *b* уместить внутри прямоугольника со сторонами *c*, *d* так, чтобы каждая из сторон одного прямоугольника была параллельна или перпендикулярна каждой стороне второго прямоугольника.

5) Дано целое число в диапазоне 100–999. Вывести строку-описание данного числа, например: 256 — «двести пятьдесят шесть», 814 — «восемьсот четырнадцать».

**Вариант 39**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны две переменные вещественного типа: *A*, *B*. Перераспределить значения данных переменных так, чтобы в *A* оказалось меньшее из значений,   
а в *B* — большее. Вывести новые значения переменных *A* и *B*.

3) Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае вычесть из него 2. Вывести полученное число.

4) Определить, войдет ли в конверт с внутренними размерами *a* и *b* мм прямоугольная открытка размером *с* и *d* мм. Для размещения открытки в конверте необходим зазор в 1 мм с каждой стороны.

5)Робот может перемещаться в четырех направлениях («С» — север, «З» — запад, «Ю» — юг, «В» — восток) и принимать три цифровые команды: 0 — продолжать движение, 1 — поворот налево, –1 — поворот направо. Дан символ *C* — исходное направление робота и целое число *N* — посланная ему команда. Вывести направление робота после выполнения полученной команды.

**Вариант 40**

1) Для данного вещественного *x* найти значение следующей функции *f*, принимающей вещественные значения:

2) Даны три числа. Найти наименьшее из них.

3) Дано целое число. Если оно является положительным, то прибавить к нему 1; если отрицательным, то вычесть из него 2; если нулевым, то заменить его на 10. Вывести полученное число.

4) Вася пытается высунуть голову в форточку размерами *a* и *b* см. Приняв условно, что его голова — круглая диаметром *d* см, определить, сможет ли Вася сделать это. Для прохождения головы в форточку необходим зазор в 1 см с каждой стороны.

5) Локатор ориентирован на одну из сторон света («С» — север, «З» — запад, «Ю» — юг, «В» — восток) и может принимать три цифровые команды поворота: 1 — поворот налево, –1 — поворот направо, 2 — поворот на 180°. Дан символ *C* — исходная ориентация локатора и целые числа *N*1 и *N*2 — две посланные команды. Вывести ориентацию локатора после выполнения этих команд.

# **Цикл с параметром: группа For**

1. Даны целые числа *K* и *N* (*N* > 0). Вывести *N* раз число *K*.

| PascalABC.NET | C++ | | |
| --- | --- | --- | --- |
| **Var**  K, N, i: Integer;  **begin**  Write('N,K: ');  Readln(N, K);  **for** i := 1 **to** N **do**  Writeln(K);  **end**. | #include <iostream>  using namespace std;  int main()  {  int k, n, i;  cin>>k>>n;  for (i = 0; i<n; ++i)  cout<<k;  system("pause");  return 0;  } | | |
| Python | C# | | |
| n= int(input("->"))  k= int(input("->"))  for i in range(1,n):  print(k) | static void Main(string[] args){  int n,k, i;  n = Convert.ToInt32(Console.ReadLine());  k = Convert.ToInt32(Console.ReadLine());  for (i = 0; i <= n; i++)  Console.WriteLine(k);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int i,n,k;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  k=sc.nextInt();  for(i=1; i<=n; i++)  System.out.print(k);  } | | | package main  import"fmt"  func main() {  fmt.Println("n,k=?")  var n,k,i int  fmt.Scanf("%d %d",&n,&k)  for i=1;i<=n; i++{  fmt.Print(k," ")  }  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** k = reader.nextInt()  **var** n = reader.nextInt()  **for** (i **in** 1..n)  *print*(**"$**k **"**) } | | | |
| Вывод четных чисел | | Обратный цикл | |
| **import** java.util.Scanner **fun** main(args: Array<String>)  {  **val** reader = Scanner  (System.*`in`*)  **var** n = reader.nextInt()  **for** (i **in** 2..n *step* 2)  *print*(**"$**i **"**) } | | **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** n = reader.nextInt()  **for** (i **in** n *downTo* 1)  *print*(**"$**i **"**) } | |
| VB.NET | | | |
| Sub Main()  Dim n, k, i As Integer  n = Convert.ToInt32(Console.ReadLine())  k = Convert.ToInt32(Console.ReadLine())  For i = 1 To n  Console.Write(k)  Next  Console.ReadKey()  End Sub | | | |
| For number As Double = 2 To 0 Step -0.25 | | | |

1. Даны два целых числа *A* и *B* (*A* < *B*). Вывести в порядке возрастания все целые числа, расположенные между *A* и *B* (включая сами числа *A* и *B*),   
   а также количество *N* этих чисел.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  A, B, i: Integer;  **begin**  Write('A,B: ');  Readln(A, B);  **for** i:= 0 **to** B - A **do**  Writeln(A + i);  Writeln(B - A + 1);  **end**. | #include <iostream>  using namespace std;  int main(){  int a, b, i;  cin>>a>>b;  for (i = a; i<=b; ++i)  cout<<i<<" ";  cout<< b-a+1;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  b= int(input("->"))  k=0;  for i in range(a,b):  print(i)  k+=1;  print('k=',k) | static void Main(string[] args){  int a, b,i, k=0;  a=Convert.ToInt32(Console.ReadLine ());  b=Convert.ToInt32 (Console.ReadLine ());  for (i = a; i <= b; i++){  k++;  Console.Write(i + " ");  }  Console.WriteLine("\n k="+k);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  int i,a,b,k=0;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  for(i=a; i<=b; i++){  k++;  System.out.print(i+" ");  }  System.out.println(k);  } | | package main  import "fmt"  func main() {  fmt.Println("a,b")  var a,b,i,k int  fmt.Scanf("%d %d", &a,&b)  k=0  for i=a; i<=b; i++ {  k++;  fmt.Println(i," ")  }  fmt.Println("\nK=",k)  } |

1. Дано вещественное число — цена 1 кг конфет. Вывести стоимость 1,   
   2, …, 10 кг конфет.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  C: Real;  i: Integer;  **begin**  Write('C: ');  Readln(C);  **for** i := 1 **to** 10 **do**  Writeln(i, ' кг: ', i \* C);  **end**. | #include <iostream>  using namespace std;  int main()  {  float c;  int i;  cin>>c;  for (i = 1; i<=10; ++i)  cout<<c\*i<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  for i in range(1,10):  print(i\*a) | static void Main(string[] args){  int a, i, n, t;  a=Convert.ToInt32 (Console.ReadLine ());  n=Convert.ToInt32 (Console.ReadLine ());  for (i = 1; i <= n; i++){  t = i \* a;  Console.Write("{0}\*{1}={2}", i,a,t);  }  Console.WriteLine();  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  int i;  double t,s;  Scanner sc=new Scanner(System.in);  t=sc.nextDouble();  for(i=1; i<=10; i++){  s=i\*t;  System.out.print(s+" ");  } | | package main  import "fmt"  func main() {  fmt.Println("t")  var i int  var s,t int  fmt.Scanf("%d",&t)  for i=1; i<=10; i++ {  s=i\* t  fmt.Println(s," ")  }  } |

1. Даны два целых числа *A* и *B* (*A* < *B*). Найти сумму всех целых чисел от *A* до *B* включительно.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B, Sum, i: Integer;  **begin**  Write('A,B: ');  Readln(A, B);  sum := 0;  **for** i := A **to** B **do**  Sum := Sum + i;  Writeln(Sum);  **end**. | | #include <iostream>  using namespace std;  int main()  {  int a,b, i, sum=0;  cin>>a>>b;  for (i =a; i<=b; ++i)  sum +=i;  cout<<sum<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("->"))  b= int(input("->"))  S=0;  for i in range(a,b):  S+=i  print('S=',S) | static void Main(string[] args){  int a, b, i, S=0;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  for (i = a; i <= b; i++){  if (i==a)  Console.Write(a);  else  Console.Write("+{0}",i);  S = S + i;  }  Console.Write("="+S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int a,b,i,s=0;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  for(i=a; i<=b; i++)  s=s+i;  System.out.print(s+" ");  } | | | package main  import "fmt"  func main() {  fmt.Println("a,b")  var i int  var s,a,b int  fmt.Scanf("%d %d",&a, &b)  for i=a; i<=b; i++ {  s=s+i;  fmt.Println(s," ")  }  } |

1. Дано целое число *N* (> 0). Найти значение выражения

1.1 – 1.2 + 1.3 – …

(*N* слагаемых, знаки чередуются). Условный оператор не использовать.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  N, i, k: Integer;  S: Real;  **begin**  Write('N: ');  Readln(N);  S := 0;  k := -1;  **for** i := 1 **to** N **do**  **begin**  k := k \* (-1);  S+=z+k\*(1+0.1\*i);  **end**;  Writeln(S);  **end**. | #include <cmath>  #include <iostream>  using namespace std;  int main()  {  int n, i;  cin>>n;  float rez=0;  for (i =1; i<=n; ++i)  rez += pow(-1.0,(i+1))\*(1.0 +  (float)i/10);  cout<<rez<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| n= int(input("->"))  h=1;  S=0;  for i in range(1,n):  S+=(1+0.1\*i)\*h  h=-h  print('S=',S) | static void Main(string[] args){  Double n, i, s = 0, h=1;  n = Convert.ToDouble(Console.ReadLine());  for (i =1.1; i <= n; i+=0.1){  s = s + i\*h;  h = -h;  }  Console.WriteLine("\nS=" +s);  Console.ReadKey();  } | |
| Java | | Go |
| public static void main(String[]args){  float s=1,t=1,i,n;  Scanner sc=new Scanner(System.in);  n=sc.nextFloat();  for(i=(float) 1.1; i<=n; i+=0.1){  s=(s+i)\*t;  t=-t;}  System.out.println(s);  } | | package main  import "fmt"  func main() {  var i,s,t,n float64  fmt.Scanf("%v",&n)  s=1  t=1  for i=1.1; i<=n; i+=0.1 {  s=(s+i)\*t;  t=-t;}  fmt.Println(s," ")  } |

1. Дано вещественное число *A* и целое число *N* (> 0). Найти *A* в степени *N*:

*AN* = *A*·*A*· … ·*A*

(числа *A* перемножаются *N* раз).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, Rez: Real;  N, i: Integer;  **begin**  Write('A,N: ');  Readln(A, N);  Rez := 1;  **for** i := 1 **to** N **do**  Rez := Rez \* A;  Writeln(Rez);  **end**. | | #include <iostream>  using namespace std;  int main()  {  float a, rez=1;  int n,i;  cin>>n>>a;  for (i =1; i<=n; ++i)  rez \*= a;  cout<<rez<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= float(input("a:"))  n= int(input("n:"))  S=1  for i in range(0,n):  S\*=a  print('S=',S) | static void Main(string[] args)  {  Double a, n, i, p = 1;  a = Convert.ToDouble(Console.ReadLine());  n = Convert.ToDouble(Console.ReadLine());  for (i = 1; i <= n; i++)  p = p \* a;  Console.WriteLine("\np=" + p);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  int i,a,n,s=1;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  n=sc.nextInt();  for(i=1; i<=n; i++)  s=s\*a;  System.out.println(s);  } | | | package main  import "fmt"  func main() {  var s,a float64  var n,i int  fmt.Scanf("%v %d", &a,&n)  s=1  for i=1; i<=n; i++ {  s=s\*a;  fmt.Println(s," ")  }  } |

1. Дано вещественное число *A* и целое число *N* (> 0). Используя один цикл, найти сумму

1 + *A* + *A*2 + *A*3 + … + *AN*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, Rez, Pow: Real;  N, i: Integer;  **begin**  Write('A,N: ');  Readln(A, N);  Rez := 1;  Pow := 1;  **for** i := 1 **to** N **do**  **begin**  Pow := Pow \* A;  Rez := Rez + Pow;  **end**;  Writeln(Rez);  **end**. | | #include <iostream>  using namespace std;  int main()  {  float a, a2=1,rez=1;  int i,n;  cin>>a>>n;  for (i =2; i<=n; ++i) {  a2\*=a;  rez +=a2;  }  cout<<rez<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= float(input("->"))  n= int(input("->"))  S=1  t=1;  for i in range(0,n):  t=t\*a  S=S+t  print('S=',S) | static void Main(string[] args){  Double a, t = 1, S=1;  int i, n;  a= Convert.ToDouble(Console.ReadLine());  n = Convert.ToInt32(Console.ReadLine());  for (i = 1; i <= n; i++){  t = t \* a;  S = S + t;  }  Console.WriteLine(S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args){  int s=1,n,i,a,t=1;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  n=sc.nextInt();  for(i=1; i<=n; i++){  s=s\*a;  t=t+s;  }  System.out.println(t);  } | | | package main  import "fmt"  func main() {  var s,a,t float64  var n,i int  fmt.Scanf("%v %d",&a,&n)  s=1  t=0  for i=1; i<=n; i++ {  s=s\*a;  t=t+s;  }  fmt.Println(t)  } |

1. Дано целое число *N* (> 0). Используя один цикл, найти сумму

1! + 2! + 3! + … + *N*!

(выражение *N*! — *N–факториал* — обозначает произведение всех целых чисел от 1 до *N*:    *N*! = 1·2·…·*N*). Чтобы избежать целочисленного переполнения, проводить вычисления с помощью вещественных переменных и вывести результат как вещественное число.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  Rez: Real;  N, i: Integer;  **begin**  Write(' N: ');  Readln(N);  Rez := 1;  **for** i := 2 **to** N - 1 **do**  **begin**  Rez := Rez + Rez \* i;  **end**;  Writeln(Rez);  **end**. | | #include <iostream>  using namespace std;  int main(){  int n, i;  float n1=1,rez=0;  cin>>n;  for (i =1; i<=n; ++i){  n1 \*=(float)i;  rez += n1;  }  cout<<rez<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("->"))  S=1  p=1;  for i in range(1,n+1):  p=p\*i  S=S+t  print('S=',S) | static void Main(string[] args){  Double S = 0, P = 1, n, i;  n=Convert.ToDouble(Console.ReadLine());  for (i = 1; i <= n; i++){  P = P \* i;  S = S + P;  }  Console.WriteLine("\nS=" + S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args){  int s=1,n,i,a,t=0;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  for(i=1; i<=n; i++){  s=s\*i;  t=t+s;  }  System.out.println(t);  } | | | package main  import "fmt"  func main() {  var t,s,n,i int  fmt.Scanf("%d",&n)  s=1  t=0  for i=1; i<=n; i++ {  s=s\*i  t=t+s  }  fmt.Println(t)  } |

1. Дано вещественное число *X* и целое число *N* (> 0). Найти значение выражения

(*N*! = 1·2·…·*N*). Полученное число является приближенным значением функции exp в точке *X*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  X, Rez, f, pow: Real;  N, i: Integer;  **begin**  Write('X,N: ');  Readln(X, N);  Rez := 1;  pow := 1;  f := 1;  **for** i := 1 **to** N **do**  **begin**  f := f \* i;  pow := pow \* X;  Rez:=Rez+pow / f;  **end**;  Writeln(Rez);  **end**. | | #include <iostream>  using namespace std;  int main()  {  float x, x1=1,n1=1,rez=1.0;  int n, i;  cin>>n>>x;  for (i =1; i<=n; ++i){  n1 \*=(float)i;  x1 \*= x;  rez += x1/n1;  }  cout<<rez<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| from math import pow  x= float(input("->"))  n= int(input("->"))  S=1;p=1  for i in range(1,n+1):  t=pow(x,i)  p=p\*i  S=S+t/p  print('S=',S) | static void Main(string[] args){  Double x, S = 1, t;  int n, P = 1, i;  x = Convert.ToDouble(Console.ReadLine());  n = Convert.ToInt32(Console.ReadLine());  for (i = 1; i <= n; i++){  t = Math.Pow(x, i);  P = P \* i;  S = S + t/ P;  }  Console.WriteLine("\nS=" + S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args){  int s=1,n,i,p=1,x;  double t=1;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  x=sc.nextInt();  for(i=1; i<=n; i++){  p=p\*x;  s=s\*i;  t=t+p/s;  }  System.out.println(t);  } | | | package main  import "fmt"  func main() {  var n,t,s,p,x,i float64  fmt.Scanf("%v %v",&n,&x)  s=1  p=1  t=0  for i=1; i<=n; i++ {  var p=p\*x  var s=s\*i  t=t+p/s  }  fmt.Println(t)  } |

1. Дано вещественное число *X* (|*X*| < 1) и целое число *N* (> 0). Найти значение выражения

.

*(полученное число является приближенным значением функции arcsin в точке X)*

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  X, Rez, pow, T1, T2: Real;  N, i: Integer;  **begin**  Write('X,N: ');  Readln(X, N);  Rez := X;  pow := X;  T1 := 1;  T2 := 1;  **for** i := 1 **to** N - 1 **do**  **begin**  T1 := T1 \* (2 \* i - 1);  T2 := T2 \* (2 \* i);  pow := pow \* X \* X;  Rez := Rez + T1 \* Pow / (T2 \* (2 \* i + 1));  **end**;  Writeln(Rez);  **end**. | #include <iostream>  using namespace std;  int main(){  double X; int N;  cin >> N>>X;  double answer;  double z, h;  double stepen;  z = stepen =1;  h = X;  answer = X;  for(int i=1; i <= N ;i++) {  // x^3 / x^5 /x^7  h = h \* X \* X;  // stepen= 1 / 3 / 15  stepen = stepen\* (2\*i-1);  // 6 / 40  z = z \* (i\*2+1)\*(i\*2);  // (1\*x^3)/6  answer += (stepen \* h) / z;  // 2 / 8  z = z / (i\*2+1);  }  cout << "answer="<<answer<<endl;  double y;  y = asin(X);  cout << "arcsin(X)= "<<y<<endl;  system("pause");  return 0;  } |
| Python | C# |
| from math import pow  x= float(input("->"))  n= int(input("->"))  S=1  p=1  for i in range(1,n+1):  p = p \* 2\*i+1;  S= S+(2\*i-1)\*pow(x,2\*i  +1)/(2\*i)\*(2\*i+1)  print('S=',S) | static void Main(string[] args)  {  Double i=0, n, x, p = 1,s=0;  n=Convert.ToDouble(Console.ReadLine());  x=Convert.ToDouble(Console.ReadLine());  for (i=1 ; i <= n; i++)  {  p = p \* 2\*i+1;  s = s+(2\*i-1)\*Math.Pow(x,2\*i+1) /  (2 \* i)\*(2 \* i + 1);  }  Console.Write(s);  Console.ReadKey();  } |
| Java | |
| public static void main(String[] args) {  Double i=0, n, x, p = 1,s=0;  Scanner sc=new Scanner(System.in);  n=sc.nextDouble();  x=sc.nextDouble();  for (i=1 ; i <= n; i++){  p = p \* 2\*i+1;  s=s+(2\*i-1)\*Math.Pow(x,2\*i+1) / (2 \* i)\*(2 \* i + 1);  }  System.out.println(s);  } | |
| Go | |
| package main  import (  "fmt"  "math")  func main() {  fmt.Println("x,n")  var n,s,p,x,i float64  fmt.Scanf("%v %v",&n,&x)  s=0  p=1  for i=1 ; i <= n; i++ {  p = p \* 2\*i+1;  s=s+(2\*i-1)\* math.Pow(x, 2 \* i + 1) / (2 \* i)\*(2 \* i + 1);  }  fmt.Println(s)  } | |

1. Дано целое число *N* (> 1) и две вещественные точки на числовой оси: *A*, *B* (*A* < *B*). Отрезок [*A*, *B*] разбит на *N* равных отрезков. Вывести *H* — длину каждого отрезка, а также набор точек

образующий разбиение отрезка [*A*, *B*].

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B, H: Real;  N, i: Integer;  **begin**  Write('N,A,B: ');  Readln(N, A, B);  H := (B - A) / N;  Writeln('H=: ', H);  **for** i := 0 **to** N **do**  Writeln(A + i \* h);  **end**. | | #include <iostream>  using namespace std;  int main(){  int n,i;  float a, b,h;  cin>>a>>b;  float h=(b-a)/n;  cout<<h<<endl;  for (i =0; i<=n; ++i){  cout<<a+i\*h;  }  system("pause");  return 0;  } | |
| Python | C# | | |
| a= float(input("->"))  b= float(input("->"))  n= int(input("->"))  h=abs(b-a)/n  for i in range(0,n+1):  print(a+i\*h) | static void Main(string[] args){  Double i, a, b, h, n=2;  a = Convert.ToDouble(Console.ReadLine());  b = Convert.ToDouble(Console.ReadLine());  h = Math.Abs(b - a) / n;  for (i = a; i <= b;i+=h )  Console.WriteLine("\ni=" + i);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  double a,h,i,n,l,b;  Scanner sc=new Scanner(System.in);  a=sc.nextDouble();  b=sc.nextDouble();  n=sc.nextDouble();  h=Math.abs(a-b)/n;  for(i=0; i<=n; i++){  l=a+i\*h;  System.out.println(l);  } | | | package main  import (  "fmt"  "math")  func main() {  fmt.Println("a,b")  var a,h,i,n,l,b float64  fmt.Scanf("%v %v %v",  &a,&b,&n)  h=math.Abs(a-b)/n;  for i=0; i<=n; i++ {  l=a+i\*h;  fmt.Println("L=",l)  }  } |

1. Дано целое число *N* (> 0). Последовательность вещественных чисел *AK* определяется следующим образом:

*A*0 = 2,        *AK* = 2 + 1/*AK*–1,    *K* = 1, 2, … .

Вывести элементы *A*1, *A*2, …, *AN*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **var**  Ai: Real;  N, i: Integer;  **begin**  Write(' N: ');  Readln(N);  Writeln('A(0)=2');  Ai := 2;  **for** i := 1 **to** N **do**  **begin**  Writeln('A(',i,')=',2+1/Ai);  Ai := 2 + 1 / Ai;  **end**;  **end**. | | #include <iostream>  using namespace std;  int main()  {  int n;  cin >> n;  double a = 2;  for (int i = 1; i <= n; ++i){  a = 2+1/a;  cout << a << " ";  }  cout << endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("->"))  a=2  print(a)  for i in range(1,n+1):  b=2+1/a  print(b)  a=b | **public** static void **Main**(string[] args) {  int n;  double a0 = 2,  a1 = 2 + 1 / a0;  n=Convert.ToInt32(Console.ReadLine());  Console.WriteLine(a0 + " " + a1);  for (int i = 2; i <= n; i++){  a0 = a1;  a1 = 2 + 1 / a0;  }  Console.WriteLine(a1);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner pa=new Scanner(System.in);  double a,b,n,i;  n=sc.nextDouble();  a=2;  System.out.println(a);  for(i=1; i<=n; i++){  b=2+1/a;  a=b;  System.out.println(b);  } | | | package main  import (  "fmt"  )  func main() {  fmt.Println("n")  var a,b,n,i float64  fmt.Scanf("%v",&n)  a=2  for i=1; i<=n; i++ {  b=2+1/a;  a=b;  fmt.Println("b=",b)  }  } |

### **Вложенные циклы**

1. Даны целые положительные числа *N* и *K*. Найти сумму

Чтобы избежать целочисленного переполнения, вычислять слагаемые этой суммы с помощью вещественной переменной и выводить результат как вещественное число.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  N, K, i, i2: Integer;  Rez, Pow: Real;  **begin**  Write('N,K: ');  Readln(N, K);  Rez := 0;  **for** i := 1 **to** N **do**  **begin**  Pow := 1;  **for** i2 := 1 **to** K **do**  Pow := Pow \* i;  Rez := Rez + Pow;  **end**;  Writeln(Rez);  **end**. | #include <iostream>  using namespace std;  int main(){  int n,k;  cin>>n>>k;  int i,i2;  float rez=0,nk;  for (i =1; i<=n; ++i) {  nk=i;  for (i2=1; i2<=k; ++i2)  nk \*=i;  rez +=nk;  }  cout <<rez<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| n= int(input("->"))  k= int(input("->"))  S=0  for i in range(1,n+1):  p=1  for j in range(1,k+1):  p=p\*i  S=S+p  print(S) | **public** static void **Main**(string[] args){  Double n, s=0, k, p = 1;  n = Convert.ToDouble(Console.ReadLine());  k = Convert.ToDouble(Console.ReadLine());  for (int i = 1; i <= n; i++){  p = 1;  for (int j = 1; j <= k; j++){  p = p \* i;  }  s = s + p;  }  Console.WriteLine("\ns=" + s);   Console.**ReadKey**(**true**); } | |
| Java | | GO |
| public static void main(String[] args) {  double j,n,i,k,s=0;  Scanner Sc=new Scanner(System.in);  n=Sc.nextDouble();  k=Sc.nextDouble();  for(i=1; i<=n; i++){  p=1;  for(j=1; j<=k; j++)  p=p\*j;  s=s+p;  }  System.out.println(s);  } | | package main  import "fmt"  func main() {  var j,n,i,k,s,p float64  fmt.Scanf("%v %v",&n, &k)  s=0  for i=1; i<=n; i++ {  p=1  for j=1; j<=k; j++ {  p=p\*j  s=s+p  }  fmt.Println("S=",s)  }  } |

1. Даны целые положительные числа *A* и *B* (*A* < *B*). Вывести все целые числа от *A* до *B* включительно; при этом каждое число должно выводиться столько раз, каково его значение (например, число 3 выводится 3 раза).

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **var**  A, B, i, i2: Integer;  **begin**  Write('A,B: ');  Readln(A, B);  **for** i := A **to** B **do**  **for** i2 := 1 **to** i **do**  Writeln(i);  **end**. | #include <iostream>  using namespace std;  int main(){  int a,b;  cin>>a>>b;  int i,i2;  for (i =a; i<=b; ++i){  for (i2=1; i2<=i; ++i2)  cout<<i<<" ";  cout<<endl;  }  system("pause");  return 0;  } | |
| Python | C# | |
| a= int(input("->"))  b= int(input("->"))  for i in range(a,b):  for j in range(0,i):  print(i) | **public** static void **Main**(string[] args) {  int a,b, i,i2;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());    for  (i =a; i<=b; ++i){      for (i2=1; i2<=i; ++i2)  Console.WriteLine(i);  Console.WriteLine();    }    Console.**ReadKey**(**true**); } | |
| Java | | GO |
| public static void main(String[] args) {  int a,b,j,i,p;  Scanner Sc=new Scanner(System.in);  a=Sc.nextInt();  b=Sc.nextInt();  for(i=a; i<=b; i++){  for(j=1; j<=i; j++)  System.out.print(i);  } | | package main  import "fmt"  func main() {  fmt.Println("k,n")  var a,b,j,i float64  fmt.Scanf("%v %v",&a, &b)  for i=a; i<=b; i++ {  for j=1; j<=i; j++ {  fmt.Println("i=",i)  }  }  } |

# **Цикл с условием: группа While**

1. Даны положительные числа *A* и *B* (*A* > *B*). На отрезке длины *A* размещено мак-симально возможное количество отрезков длины *B* (без наложений). Не исполь-зуя операции умножения и деления, найти длину незанятой части отрезка *A*.

| PascalABC.NET | C++ | | |
| --- | --- | --- | --- |
| **Var**  A, B, h: Integer;  **begin**  Write('A,B: ');  Readln(A, B);  temp := A;  **while** temp >= 0 **do**  h := h - B;  Writeln(h + B);  **end**. | #include <iostream>  using namespace std;  int main(){  int a,b;  cin>>a>>b;  int h=a;  while (h>=0){  h -= b;  }  cout<<h+b;  system("pause");  return 0;  } | | |
| Python | C# | | |
| a= int(input("->"))  b= int(input("->"))  while (a>=b):  a=a-b  print(a) | **public** static void **Main**(string[] args){  int a, b;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  while (a >= b){  a = a - b;  Console.WriteLine(a + " ");  }  Console.WriteLine("\na=" + a);    Console.**ReadKey**(**true**); } | | |
| Java | | | Go |
| public static void main(String[]args){  int a,b;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  while(a>=b){  a=a-b;  }  System.out.println(a);  } | | | package main  import "fmt"  func main() {  fmt.Println("a,b=?")  var a,b int  fmt.Scanf("%d %d", &a,&b)  for a>b{  a=a-b  }  fmt.Println(a)  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a = reader.nextInt()  **var** b = reader.nextInt()  **while** (a >= b)  a -= b  *print*(a) } | | **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** a = reader.nextInt()  **var** b = reader.nextInt()  **do** {  a -= b  }**while**(a>b)  *print*(a) } | |

1. Дано целое число *N* (> 0). Если оно является степенью числа 3, то вывести True, если не является — вывести False.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, temp: Integer;  **begin**  Write('A: ');  Readln(A);  temp := A;  **while** temp < A **do**  temp := temp\*3;  Write(temp = A);  **end**. | | #include <iostream>  using namespace std;  int main()  {  int a;  cin>>a;  while (a%3==0)  a /=3;  cout<<(a==1)?"True":"False";  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("->"))  while (n%3==0):  n/=3  print(n==1) | **public** static void **Main**(string[] args){  int n;  bool s;  n = Convert.ToInt32(Console.ReadLine());  while (n%3==0){  n = n / 3;  }  s = n == 1;  Console.WriteLine(s);   Console.**ReadKey**(**true**); } | | |
| Java | | | GO |
| public static void main(String[] args) {  boolean y;  int n;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  while(n%3==0){  n=n/3;  }  y=n==1;  System.out.println(y);  } | | | package main  import "fmt"  func main() {  var n int  var y bool  fmt.Scanf("%v",&n)  for n%3==0 {  n=n/3;  }  y=n==1;  fmt.Println("y=",y)  } |
| VB.NET | | | |
| Варинат 1.  Sub Main()  Dim a, b As Integer  a = Convert.ToInt32(Console.ReadLine())  b = Convert.ToInt32(Console.ReadLine())  While a >= b  a -= b  End While  Console.WriteLine(a)  Console.ReadKey()  End Sub  Вариант 2  Sub Main()  Dim a, b As Integer  a = Convert.ToInt32(Console.ReadLine())  b = Convert.ToInt32(Console.ReadLine())  Do While a >= b  a -= b  Loop  Console.WriteLine(a)  Console.ReadKey()  End Sub  Вариант 3.  Sub Main()  Dim a, b As Integer  a = Convert.ToInt32(Console.ReadLine())  b = Convert.ToInt32(Console.ReadLine())  Do Until a <= b  a -= b  Loop  Console.WriteLine(a)  Console.ReadKey()  End Sub | | | |

1. Дано целое число *N* (> 0), являющееся некоторой степенью числа 2: *N* = 2*K*. Найти целое число *K* — показатель этой степени.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  N, K: Integer;  **begin**  Write(' N: ');  Readln(N);  K := 0;  **while** n > 1 **do**  **begin**  n := n **div** 2;  inc(K);  **end**;  Writeln('K: ', K);  **end**. | #include <iostream>  using namespace std;  int main()  {  int n;  cin>>n;  int k=0;  while (n>=2){  ++k;  n/=2;  }  cout<<k<<endl;  system("pause");  return 0;  } | |
| Python | C# | |
| n= int(input("a: "))  p=2  k=1  while (p<n):  p=p\*2  k+=1  print(k) | **public** static void **Main**(string[] args) {  int n, k=0;  n = Convert.ToInt32(Console.ReadLine());  while (n >= 2){  n = n / 2;  k++;  }  Console.WriteLine(k);  Console.**ReadKey**(**true**); } | |
| Java | | GO |
| public static void main(String[] args) {  int n,k=1;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  while(n>2){  k++;  n=n/2;  }  System.out.println(k);  } | | package main  import "fmt"  func main() {  fmt.Println("n->!")  var k,n int  fmt.Scanf("%d",&n)  k=1  for n>2 {  k++;  n=n/2;  }  fmt.Println(k)  } |

1. Дано целое число *N* (> 0). Найти наибольшее целое число *K*, квадрат которого не превосходит *N*: *K*2*N*. Функцию извлечения квадратного корня не использовать.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **Var**  N, k: Integer;  **begin**  Write('N: ');  Readln(N);  K := 1;  **while** Sqr(K) <= N **do**  inc(K, 1);  Writeln(K - 1);  **end**. | #include <iostream>  using namespace std;  int main()  {  int n;  cin>>n;  int k=1;  while (k\*k<=n)  ++k;  cout<<k-1;  system("pause");  return 0;  } | |
| Python | C# | |
| n= int(input("n:"))  k=1  while (k\*k<=n):  k+=1  print(k-1) | **public** static void **Main**(string[] args) {  Double n, k = 0, t;  n = Convert.ToInt32(Console.ReadLine());  while ((k \* k)<= n){  k++;  }  t=(k\*k)>n ? k-1:k;  Console.WriteLine(t);  Console.ReadKey(); } | |
| Java | | GO |
| public static void main(String[] args) {  int n,k=0,p = 0;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  while(k\*k<=n){  k++;  p=k\*k;  }  k=p>n?k-1:k;  System.out.println(k+" ");  } | | package main  import "fmt"  func main() {  fmt.Println("n->")  var k,n int  fmt.Scanf("%d",&n)  k=1  for k\*k<=n {  k++;  }  fmt.Println(k-1)  } |

1. Дано целое число *N* (> 1). Вывести наибольшее из целых чисел *K*, для которых сумма 1 + 2 + … + *K* будет меньше или равна *N*, и саму эту сумму.

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **var**  N, K, Temp: Integer;  **begin**  Write('N: ');  Readln(N);  K := 0;  Temp := 0;  **while** Temp < N **do**  **begin**  inc(K, 1);  Temp := Temp + K;  **end**;  Writeln(K, ' ', Temp);  **end**. | #include <iostream>  using namespace std;  int main()  {  int N;  int K= 0;  int S= 0;  cout << " N (>1): ";  cin >> N;  while(S <= N) {  ++K;  S += K;  }  cout <<"K= " << K << endl;  system("pause");  return 0;  } | |
| Python | C# | |
| n= int(input("n: "))  S=0  k=0  while (S<n):  k+=1  S+=k  print(k) | **public** static void **Main**(string[] args) {  Double n, p=0, k=0;  n = Convert.ToInt32(Console.ReadLine());  while (p < n){  k++;  p= p + k;  }  Console.WriteLine(k);   Console.**ReadKey**(**true**); } | |
| Java | | GO |
| public static void main(String[] args) {  int n,k=0,p = 0,s=0;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  while(s<n){  k++;  s=s+k;  }  System.out.println(k+" "+s);  } | | package main  import (  "fmt"  )  func main() {  fmt.Println("n")  var k,n,s int  fmt.Scanf("%d",&n)  s=0  k=0  for s<n {  k++;  s=s+k;  }  fmt.Println(k," ",s)  } |

1. Начальный вклад в банке равен 1000 руб. Через каждый месяц размер вклада увеличивается на *P* процентов от имеющейся суммы (*P* — вещественное число, 0 < *P* < 25). По данному *P* определить, через сколько месяцев размер вклада превысит 1100 руб., и вывести найденное количество месяцев *K* (целое число) и итоговый размер вклада *S* (вещественное число).

| PascalABC.NET | C++ | |
| --- | --- | --- |
| **var**  P, S: Real;  K: Integer;  **begin**  Write('p: ');  Readln(P);  K := 1;  S := 1000;  **while** S <= 1100 **do**  **begin**  inc(K, 1);  S := S + S \* P / 100;  **end**;  Writeln(K, ' ', S);  **end**. | #include <iostream>  using namespace std;  int main()  {  float p;  cin>>p;  int k=1;  float s=1000;  while (s<=1100) {  ++k;  s+=s\*p/100;  }  cout<<"K="<<k<<" S="<<s<<"\n";  system("pause");  return 0;  } | |
| Python | C# | |
| p= float(input("->"))  S=1000.0  k=1  while (S<1100):  k+=1  S+=(S\*p)/100;  print(k) | **public** static void **Main**(string[] args) {  float p;  p = Convert.ToDouble(Console.ReadLine());  int k=1;  float s=1000;  while (s<=1100) {      ++k;      s+=s\*p/100;  }  Console.WriteLine(k+” “+s);  Console.**ReadKey**(**true**); } | |
| Java | | GO |
| public static void main(String[] args) {  double p,k=0,s=1000;  Scanner sc=new Scanner(System.in);  p=sc.nextDouble();  while(s<=1100){  k++;  s=s+1000\*p/100;  }  System.out.println(k+" "+s);  } | | package main  import (  "fmt"  )  func main() {  fmt.Println("p")  var k,p,s float64  fmt.Scanf("%v",&p)  s=1000  k=0  for s<=1100 {  k++;  s=s+1000\*p/100;  }  fmt.Println(k," ",s)  } |

1. Дано целое число *N* (> 0). Используя операции деления нацело и взятия остатка от деления, вывести все его цифры, начиная с самой правой (разряда единиц).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  N: Integer;  **begin**  Write('N: ');  Readln(N);  **while** N > 0 **do**  **begin**  Writeln(N **mod** 10);  N := N **div** 10;  **end**;  **end**. | | #include <iostream>  using namespace std;  int main()  {  int n;  cin>>n;  while (n>0) {  cout<<n%10;  n /=10;  }  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("->"))  while (n>0):  print(n%10)  n//=10 | **public** static void **Main**(string[] args) {   int n;  n = Convert.ToInt32(Console.ReadLine());  int k=1;  while (n>0) {  Console.WriteLine(n%10);       n /=10;     }   Console.**ReadKey**(); } | | |
| Java | | | GO |
| public static void main(String[] args) {  int n,t,s=0;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  while(n!=0){  t=n%10;  s=s\*10+t;  n=n/10;  System.out.print(t+" ");  }  } | | | package main  import "fmt"  func main() {  fmt.Println("n")  var n,t,s int  fmt.Scanf("%d",&n)  s=0  for n!=0 {  t=n%10;  s=s\*10+t;  n=n/10;  fmt.Println(t," ")  }  }  } |

1. Дано целое число *N* (> 0). С помощью операций деления нацело и взятия остатка от деления определить, имеется ли в записи числа *N* цифра «2». Если имеется, то вывести True, если нет — вывести False.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  N: Integer;  z: Boolean;  **begin**  Write('N: ');  Readln(N);  z := False;  **while** (N>0) **and** (**not** z) **do**  **begin**  **if** (N **mod** 10) = 2 **then**  z := True;  N := N **div** 10;  **end**;  Writeln(z);  **end**. | | #include <iostream>  using namespace std;  int main()  {  int n;  cin>>n;  while((n>0)&((n%10)!=2)) {  n /=10;  }  cout<<((n>0)&((n%10)==2));  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("->"))  while ((n%10)%2==0):  n//=10  print((n%10)%2==1) | **public** static void **Main**(string[] args){  int n;  n = Convert.ToInt32(Console.ReadLine());  while((n>0)&((n%10)!=2)) {         n /=10;     }  Console.WriteLine( ((n>0)&(n%10)==2));    Console.**ReadKey**(); } | | |
| Java | | | GO |
| public static void main(String[] args) {  int n,t;  boolean y;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  while(n!=0){  t=n%10;  n=n/10;  y=(t==2);  System.out.print(y+" ");  }  } | | | package main  import "fmt"  func main() {  fmt.Println("n")  var n,t int  var y bool  fmt.Scanf("%d",&n)  for n!=0 {  t=n%10;  n=n/10;  y=(t==2);  fmt.Println(y," ")  }  } |

1. Даны целые положительные числа *A* и *B*. Найти их *наибольший общий делитель* (НОД), используя *алгоритм Евклида*:

НОД(*A*, *B*) = НОД(*B*, *A* mod *B*),    если ;        НОД(*A*, 0) = *A*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B: Integer;  **begin**  Write('A,B: ');  Readln(A, B);  **while** (A <> 0) **and** (B <> 0) **do**  **begin**  **if** a > b **then** A := (a **mod** b)  **else** B := (B **mod** A);  **end**;  Writeln(A + B);  **end**. | | #include <iostream>  using namespace std;  int main(){  int a,b;  cin>>a>>b;  while ((a!=0)&(b!=0)){  if (a>b) a=a%b;  else b=b%a;  }  cout<<a+b<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= int(input("a: "))  b= int(input("b: "))  while ((a!=0) and (b!=0)):  if (a>b):  a=a%b  else:  b=b%a  print(a+b) | **public** static void **Main**(string[] args){  int a,b;  a = Convert.ToInt32(Console.ReadLine());  b = Convert.ToInt32(Console.ReadLine());  while ((a!=0)&(b!=0)){      if (a>b)  a=a%b;      else  b=b%a;     }  Console.WriteLine( a+b);   Console.**ReadKey**(); } | | |
| Java | | | GO |
| public static void main(String[] args) {  int a,b;  Scanner sc=new Scanner(System.in);  a=sc.nextInt();  b=sc.nextInt();  while(a!=b){  if(a>b)  a=a-b;  else  b=b-a;  }  System.out.println(a+" ");  } | | | package main  import "fmt"  func main() {  fmt.Println("n")  var a,b int  fmt.Scanf("%d %d",&a,&b)  for a!=b {  if a>b {  a=a-b;  }else {  b=b-a;  }  fmt.Println("a=",a)  }  } |

1. Дано целое число *N* (> 1). Последовательность *чисел Фибоначчи* *FK* определяется следующим образом:

Проверить, является ли число *N* числом Фибоначчи. Если является, то вывести True, если нет — вывести False.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **var**  N, F, F1, F2: Integer;  **begin**  Write('N: ');  Readln(N);  F1 := 1;  F2 := 1;  F := 0;  **while** F < N **do**  **begin**  F := F2 + F1;  F2 := F1;  F1 := F;  **end**;  Writeln(F = N);  **end**. | | #include <iostream>  using namespace std;  int main()  {  int n;  cin>>n;  int f1=1,f2=1,f=0;  while (f<n){  f=f2+f1;  f2=f1;  f1=f;  }  cout<<(f==n)?"True":"False";  system("pause");  return 0;  } | |
| Python | C# | | |
| n= int(input("->"))  f1=1  f2=1  f=0  while (f<n):  f=f2+f1;  f1=f2;  f2=f  print(f==n) | **public** static void **Main**(string[] args) {    int n, f1=1,f2=1,f=0;;  n=Convert.ToInt32(Console.ReadLine());  while (f<n){         f=f2+f1;         f1=f2;         f2=f;     }  Console.WriteLine(f==n?"True":"False");     Console.**ReadKey**(); } | | |
| Java | | | GO |
| public static void main(String[] args) {  int f1,f2,f,n;  boolean y=false;  Scanner sc=new Scanner(System.in);  n=sc.nextInt();  f1=1;  f2=1;  f=f1+f2;  while(f<=n){  f1=f2;  f2=f;  f=f1+f2;  if(f>n)  y=true;  }  System.out.println(f+" ");  } | | | package main  import "fmt"  func main() {  fmt.Println("n")  var f1,f2,f,n int  fmt.Scanf("%d",&n)  f1=1;  f2=1;  f=f1+f2;  for f<=n {  f1=f2;  f2=f;  f=f1+f2;  fmt.Println(f," ")  }  } |

1. Дано вещественное число Последовательность вещественных чисел *AK* определяется следующим образом:

.

Найти первый из номеров *K*, для которых выполняется условие

|*AK* – *AK*–1| < ,

и вывести этот номер, а также числа *AK*–1 и *AK*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  k: Integer;  Ak1, Ak, e: Real;  **begin**  Write('e: ');  Readln(e);  Ak1 := 0;  Ak := 2;  K := 1;  **while** Abs(Ak-Ak1)>=e **do**  **begin**  Inc(k, 1);  Ak1 := Ak;  Ak := 2 + 1 / Ak1;  **end**;  Writeln(k, ' ', Ak1, '', Ak);  **end**. | | #include <iostream>  using namespace std;  int main(){  float e; cin>>e;  float ak1=0,ak=2;  int k=1;  while (fabs(ak-ak1)>=e) {  ++k;  ak1=ak;  ak=2+1/ak1;  }  cout<<k<<ak1<<ak<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| e= float(input("e:"))  k=2  a1=2  a2=2+1.0/a1;  h=abs(a2-a1);  while (h>=e):  k+=1  a1=a2;  a2=2+1.0/a1;  h=abs(a2-a1);  print(k,' ',a1,' ',a2) | **public** static void **Main**(string[] args){  double e, ak1=0,ak=2;  e= Convert.ToSingle(Console.ReadLine());  int k=1;   while (fabs(ak-ak1)>=e) {       ++k;       ak1=ak;       ak=2+1/ak1;    }  Console.WriteLine(k+” “+ak1+” “+ak);   Console.**ReadKey**(); } | | |
| Java | | | GO |
| public static void main(String[] args) {  double a1,a2, h,eps;  int k=2;  Scanner sc=new Scanner(System.in);  eps=sc.nextDouble();  a1=2;  a2=2+1.0/a1;  h=Math.abs(a2-a1);  while (h>=eps){  k++;  a1=a2;  a2=2+1.0/a1;  h=Math.abs(a2-a1);  }  System.out.println("k="+k+" a1="+  a1+" a2="+a2);  } | | | package main  import (  "fmt"  "math")  func main() {  var a1,a2, h,eps float64  var k int  fmt.Scanf("%v",&eps)  a1=2;  k=2;  a2=2+1.0/a1;  h=math.Abs(a2-a1);  for h>=eps {  k++;  a1=a2;  a2=2+1.0/a1;  h=math.Abs(a2-a1);  }  fmt.Println("k=",k,"\na1=",  a1, "\na2=",a2)  } |

1. Даны положительные числа *A*, *B*, *C*. На прямоугольнике размера размещено максимально возможное количество квадратов со стороной *C* (без наложений). Найти количество квадратов, размещенных на прямоугольнике. Операции умножения и деления не использовать.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A, B,H, C: Real;  K: integer;  **begin**  Readln(A, B, C);  K := 0;  **while** (A - C >= 0) **do**  **begin**  A := A - C;  H := B;  **while** (H - C >= 0) **do**  **begin**  H := H - C;  inc(K, 1);  **end**;  **end**;  Writeln(k);  **end**. | | #include <iostream>  using namespace std;  int main(){  float a,b,c;  cin>>a>>b>>c;  int k=0;  float temp;  while ((a-c)>=0)  {  a-=c;  temp=b;  while (temp-c>=0)  {  temp-=c;  ++k;  }  }  cout<<k<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| a= float(input("a: "))  b= float(input("b: "))  c= float(input("c: "))  k=0  h=0  while ((a-c)>=0):  a-=c  h=b  while (h-c>=0):  h-=c  k+=1  print(k) | **public** static void **Main**(string[] args) {  float a,b,c;  a=Convert.ToSingle(Console.ReadLine());  b=Convert.ToSingle(Console.ReadLine());  c=Convert.ToSingle(Console.ReadLine());    int k=0;    float temp;    while ((a-c)>=0) {        a-=c;         temp=b;         while (temp-c>=0){             temp-=c;             ++k;         }    }  Console.WriteLine(k);    Console.**ReadKey**(); } | | |
| Java | | | GO |
| public static void main(String[]args)  {  double A,B,C,h=0,k=0,s=0,i=1;  Scanner sc=new Scanner(System.in);  A=sc.nextDouble();  B=sc.nextDouble();  C=sc.nextDouble();  while(A>=C){  A=A-C;  k++;}  while(B>=C){  B=B-C;  h++;  }  while(i<=k){  s=s+h;  i++;  }  System.out.println(s+" ");  }  } | | | package main  import "fmt"  func main()  {  fmt.Println("n")  var A,B,C,h,k,s,i float64  fmt.Scanf("%v %v %v",  &A,&B,&C)  h=0; k=0; s=0; i=1;  for A>=C {  A=A-C;  k++; }  for B>=C {  B=B-C;  h++;  }  for i<=k {  s=s+h;  i++;  }  fmt.Println(s," ")  } |

# **Последовательности: группа Series**

Во всех заданиях данной группы предполагается, что исходный набор содержит ненулевое число элементов (в частности, число *N* всегда больше нуля). В заданиях на обработку нескольких наборов чисел количество наборов *K* также всегда является ненулевым.

Для решения заданий из данной группы следует использовать «однопроходные» алгоритмы, позволяющие получить требуемый результат после *однократного* просмотра набора исходных данных. Однопроходные алгоритмы обладают важным преимуществом: для них не требуется хранить в памяти одновременно весь набор данных, поэтому при программной реализации этих алгоритмов *можно* **не использовать массивы.**

1. Даны десять вещественных чисел. Найти их сумму.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  R, Rez: Real;  i: Integer;  **begin**  Rez := 0;  **for** i := 1 **to** 10 **do**  **begin**  Readln(R);  Rez := Rez + R;  **end**;  Writeln(Rez);  **end**. | | int main()  {  float S = 0, r;  for (int i = 1; i <= 10; ++i) {  cin >> r;  S+= r;  }  cout << S;  return 0;  } | |
| Python | C# | | |
| S=0  for i in range(1,10):  a=float(input("->"))  S=S+a  print(S) | static void Main(string[] args)  {  float a, S = 0;  for (int i=0; i<10; i++){  a=Convert.ToSingle(Console.ReadLine());  S = S + a;  }  Console.WriteLine(S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args){  Scanner sc=new Scanner(System.in);  float a, S = 0;  for (int i=0; i<10; i++){  a =sc.nextFloat();  S+=a;  }  System.out.println(S);  } | | | package main  import "fmt"  func main() {  var S=0.0  var a=0.0  for i:=1; i<=10; i++{  fmt.Scan(&a)  S=S+a  }  fmt.Println(S)  } |
| Kotlin | | | |
| import java.util.Scanner fun main(args: Array<String>){  var S: Double = 0.0  val reader = Scanner(System.`in`)  for(i in 1..10){  val x = reader.nextDouble()  S=S+x;  }  println(S) } | | | |
| VB.NET | | | |
| Sub Main()  Dim a, S As Single  For i As Integer = 1 To 10  a = Convert.ToSingle(Console.ReadLine())  S += a  Next  Console.WriteLine(S)  Console.ReadKey()  End Sub | | | |

1. Дано целое число *N* и набор из *N* положительных вещественных чисел. Вывести в том же порядке целые части всех чисел из данного набора (как вещественные числа с нулевой дробной частью), а также сумму всех целых частей.

| PascalABC.NET | C++ |
| --- | --- |
| **Begin**  write('N = ');  **var** N := readInteger;  **var** S: real:= 0;  **for var** i := 1 **to** N **do**  **begin**  **var** a := readReal;  S := S + int(a);  write(int(a), ' ')  **end**;  writeln;  writeln(S);  **end**. | int main()  {  int S = 0, i;  float r;  for (i = 1; i <= 10; ++i) {  cin >> r;  S+= (int)r;  cout << (int)r << endl;  }  cout << S;  return 0;  } |
| Python | |
| S=0  n=int(input("->"))  for i in range(0,n):  a=float(input("->"))  S=S+int(a)  print(int(a))  print(S) | |

1. Даны целые числа *K*, *N* и набор из *N* целых чисел. Если в наборе имеются числа, меньшие *K*, то вывести True; в противном случае вывести False.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  check:Boolean;  i,N,K,R:Integer;  **begin**  (N,K):=ReadInteger2;  check:=False;  **for** i:=1 **to** N **do**  **begin**  Readln(R);  **if** R<K **then**  check:=True;  **end**;  Writeln(check);  **End** | int main()  {  int k, n, i, r;  bool b = 0;  cin >> n >> k;  for (i = 1; i <= n; i++) {  cin >> r;  if (r<k)  b = 1;  }  cout<<b<<endl;  system("pause");  return 0;  } |

1. Дано целое число *K* и набор ненулевых целых чисел; признак его завершения — число 0. Вывести количество чисел в наборе, меньших *K*.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  i, R, K: Longint;  **begin**  readln(K);  R := 1;  i := 0;  **while**(R <> 0) **do**  **begin**  Readln(R);  **if** R < K **then**  i += 1;  **end**;  i -= 1;  Writeln(i);  **end**. | #include <iostream>  using namespace std;  int main(){  int a, i = -1, k;  bool y = 0;  cin >> k;  do {  i++;  cin >> a;  if (a > k)  y = 1;  } while (a != 0);  cout<<y<<" "<<i<<endl;  system("pause");  return 0;  } |

1. Дано вещественное число *B*, целое число *N* и набор из *N* вещественных чисел, упорядоченных по возрастанию. Вывести элементы набора вместе с числом *B*, сохраняя упорядоченность выводимых чисел.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  n, i, c: integer;  b, num: real;  **begin**  read(n);  read(b);  c := 0;  **for** i := 1 **to** n **do**  **begin**  readln(num);  **if** (c=0) **and** (b<num) **then**  **begin**  writeln(b);  c := 1;  **end**;  writeln(num);  **end**;  **if** c = 0 **then** write(b);  **end**. | int main(){  int n;  float b,r=1.0;  cout<<"N, B";  cin >> n >> b;  bool f=1;  for (int i=1; i<=n; i++){  cin>>r;  if ((b<=r)&&f){  cout<<b;  f=0;  }  cout<< r;  system("pause");  return 0;  } |

1. Дано целое число *N* (> 1) и набор из *N* целых чисел. Вывести те элементы в наборе, которые меньше своего левого соседа, и количество *K* таких элементов.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  i, K, N, Num, Num1: Integer;  **begin**  Readln(N);  K := 0;  **for** i := 1 **to** N **do**  **begin**  Readln(Num);  **if** (Num1>Num) **and** (i<>1) **then**  **begin**  writeln(Num);  inc(K);  **end**;  Num1 := Num;  **end**;  Writeln(K);  **end**. | int main(){  int n, i, a,b, kol=0;  cin>>n>>b;  for (i=1; i<=n; i++){  cin >> a;  if (a < b) {  cout << a << "\n";  kol++;  }  b = a;  }  cout << kol << endl;  system("pause");  return 0;  } |

1. Дано целое число *N* (> 1) и набор из *N* вещественных чисел. Проверить, образует ли данный набор возрастающую последовательность. Если образует, то вывести True, если нет — вывести False.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a, Prev: real;  N, i: byte;  IncSeq: boolean;  **begin**  write('N = ');  readln(N);  IncSeq := true;  **for** i := 1 **to** N **do**  **begin**  read(a);  **if** i > 1 **then**  **if** IncSeq **then**  **if not** (a > Prev) **then**  IncSeq := false;  Prev := a  **end**;  writeln;  writeln(IncSeq);  **end**. | #include <iostream>  using namespace std;  int main(){  int n, i, a,b;  bool y = 1;  cin>>n>>b;  for (i=2; i<=n; i++){  cin >> a;  if (b>a)  y = 0;  b = a;  }  cout << y << endl;  system("pause");  return 0;  } |

1. Дано целое число *N* (> 2) и набор из *N* вещественных чисел. Набор называется *пилообразным*, если каждый его внутренний элемент либо больше, либо меньше обоих своих соседей (то есть является «зубцом»). Если данный набор является пилообразным, то вывести 0; в противном случае вывести номер первого элемента, не являющегося зубцом.

| PascalABC.NET |
| --- |
| **Var**  a, Left, Right: real;  N, i, Number: byte;  **begin**  write('N = ');  readln(N);  Number := 0;  **for** i := 1 **to** N **do**  **begin**  read(Right);  **if** i = 1 **then**  Left := Right  **else**  **if** i = 2 **then**  a := Right  **else begin**  **if** Number = 0 **then**  **if not** (((a < Right) **and** (a < Left)) **or**  ((a > Right) **and** (a > Left))) **then**  Number := i;  Left := a;  a := Right  **end**  **end**;  writeln;  **if** Number = 0 **then**  writeln('Последовательность пилообразная')  **else**  writeln('Номер элемента, нарушающего закономерность:',Number)  **end**. |
| C++ |
| #include <iostream>  using namespace std;  bool pilob(int a, int b, int c) {  return ((a > b) && (b < c)) || ((a < b) && (b > c));  }  int main(){  int n, i, a, b, h;  bool y= 1;  cin>>n>>a>>h;  for (i=3; i<=n; i++){  cin >> b;  if (!pilob(a, h, b))  y=0;  cout << a << " " << h << " " << b << endl;  a = h;  h = b;  }  cout << y << endl;  system("pause");  return 0;  } |

### **Вложенные циклы**

1. Даны целые числа *K*, *N* и набор из *N* вещественных чисел: *A*1, *A*2, …, *AN*. Вывести *K*-e степени чисел из данного набора:

(*A*1)*K*, (*A*2)*K*, …, (*AN*)*K*.

| PascalABC.NET |
| --- |
| **var**  K, N, i, j: byte;  A, S: real;  **begin**  write('K = '); readln(K);  write('N = '); readln(N);  **for** i := 1 **to** N **do**  **begin**  writeln;  write('A', i, ' = ');  readln(A);  S := 1;  **for** j := 1 **to** K **do**  S := S \* A;  writeln(' --> ', A, '^', K, ' = ', S)  **end**;  **end**. |
| C++ |
| int main(){  int k, n;  cin >> n >> k;  int i, j;  float a, P;  for (i = 1; i <= n; ++i) {  cin >> a;  P = 1;  for (j = 1; j <= k; j++)  P \*= a;  cout << P << endl;  }  system("pause");  return 0;  } |

1. Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Вывести общую сумму всех элементов, входящих в данные наборы.

| PascalABC.NET |
| --- |
| **var**  K, N, Ni, Ki, A, S: Integer;  **begin**  Write('N: ');  Readln(N);  Write('K: ');  Readln(K);  S := 0;  **for** Ki := 1 **to** K **do**  **for** Ni := 1 **to** N **do**  **begin**  write('Введите ', Ni, ' элемент из ', Ki, ' набора :');  Readln(A);  S := S + A;  **end**;  writeln('Сумма элементов равна: ', S);  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main(){  int k, n, i, j,a, S=0;  cin >> k >> n;  for (i = 1; i <= k; ++i)  for (j = 1; j <= n; j++){  cin >> a;  S += a;  }  cout<<S<<endl;  system("pause");  return 0;  } |

11) Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Для каждого набора вывести сумму его элементов.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  a, Sum: integer;  K, N, i, j: byte;  **begin**  write('K = ');  readln(K);  write('N = ');  readln(N);  **for** i := 1 **to** K **do**  **begin**  writeln;  Sum := 0;  **for** j := 1 **to** N **do**  **begin**  read(a);  Sum := Sum + a  **end**;  writeln(' S= ', Sum)  **end**;  **end**. | #include <iostream>  using namespace std;  int main(){  int k, n, i, j,a, S=0;  cin >> k >> n;  for (i = 1; i <= k; ++i) {  S = 0;  for (j = 1; j <= n; j++) {  cin >> a;  S += a;  }  cout << "S="<<S << endl;  }  system("pause");  return 0;  } |

1. Дано целое число *K*, а также *K* наборов ненулевых целых чисел. Признаком завершения каждого набора является число 0. Для каждого набора вывести количество его элементов. Вывести также общее количество элементов во всех наборах.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  a, Q, count: integer;  K, i: byte;  **begin**  write('K = ');  readln(K);  Q := 0;  **for** i := 1 **to** K **do**  **begin**  count := 0;  **repeat**  read(a);  inc(count)  **until** a = 0;  writeln('--> ', count);  Q := Q + count  **end**;  writeln('Q= ', Q);  **end**. | #include <iostream>  using namespace std;  int main(){  int n, i, j,a, S=0, kol;  cin >> n;  for (i = 1; i <= n; ++i) {  kol = 0;  cin >> a;  while (a!=0){  kol++;  cin >> a;  }  cout << "kol=" << kol << endl;  S += kol;  }  cout << S << endl;  system("pause");  return 0;  } |

1. Дано целое число *K*, а также *K* наборов ненулевых целых чисел. Каждый набор содержит не менее двух элементов, признаком его завершения является число 0. Для каждого набора выполнить следующее действие: если элементы набора возрастают, то вывести 1; если элементы набора убывают, то вывести –1; если элементы набора не возрастают и не убывают, то вывести 0.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  a, b, K, i, p: integer;  **begin**  readln(K);  **for** i := 1 **to** K **do**  **begin**  read(b);  read(a);  **if** a < b **then** p := -1  **else**  **if** a > b **then** p := 1  **else** p := 0;  **while** a <> 0 **do**  **begin**  **if** p <> 0 **then**  **if** (p = -1) **and** (a < b)  **or** (p=1) **and** (a>b) **then**  b := a  **else** p := 0;  read(a)  **end**;  writeln(p)  **end**;    **end**. | #include <iostream>  using namespace std;  int main(){  int k;  cin >> k;  int count=0,i,a=1,b;  for (i=1; i<=k; ++i){  cout<<">";  cin>>a;  int checkup=0,  checkdown=0;  while(a!=0){  b=a;  cout<<">";  cin>>a;  if ((b<=a)&&(a!=0))  checkup=1;  if ((b>=a)&&(a!=0))  checkdown=1;  }  if (checkup&checkdown)  cout << "0" << endl;  else if (checkup)  cout << "1" << endl;  else if (checkdown)  cout << "-1" << endl;  }  system("pause");  return 0;  } |

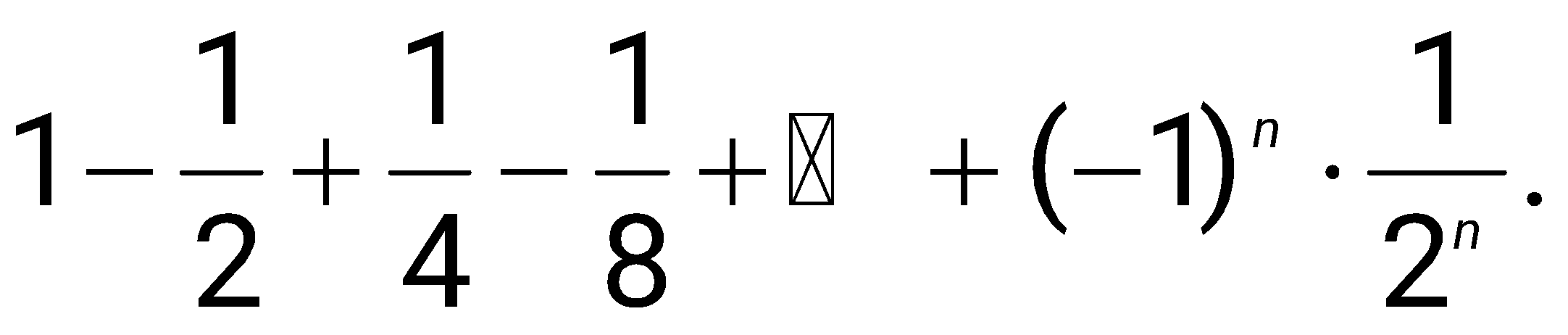
**Задачи для самостоятельной работы**

***Вариант 1.***

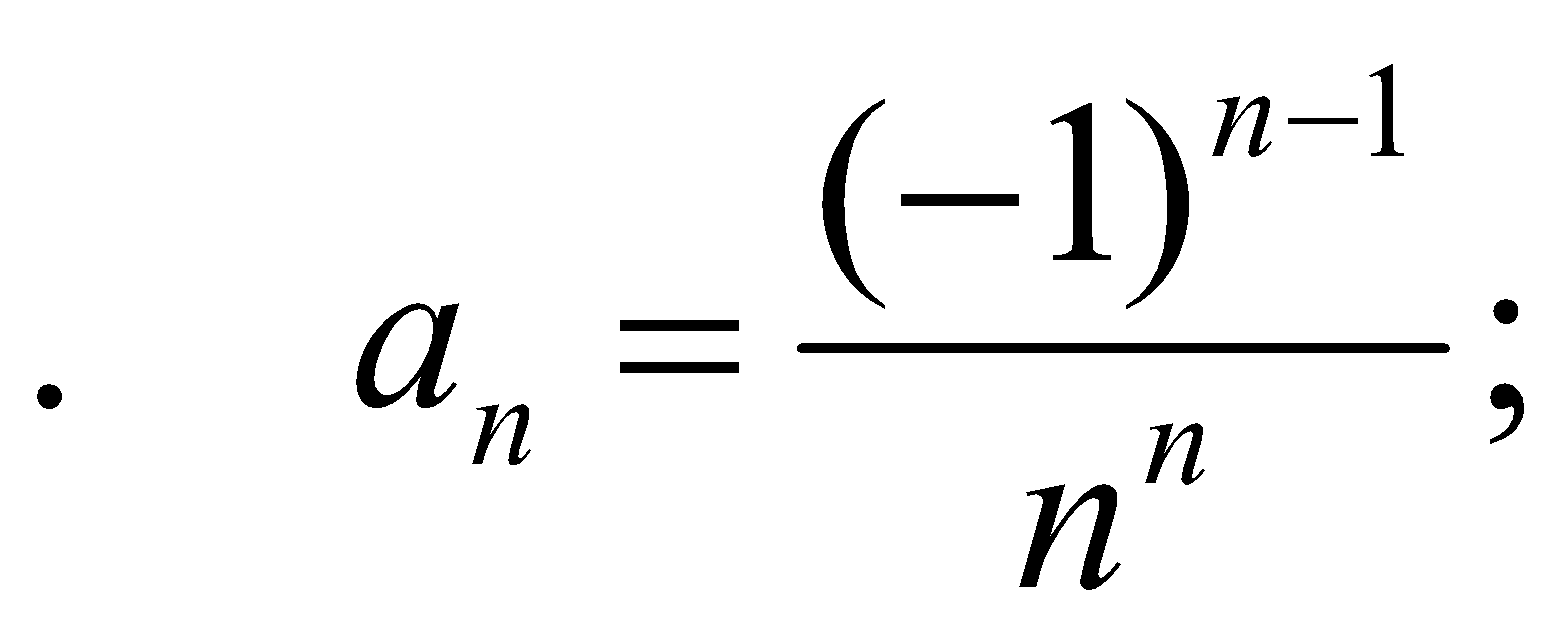
1. Имеется серия измерений элементов треугольника. Группы элементов пронумерованы. В серии в произвольном порядке могут встречаться такие группы элементов треугольника:

* основание и высота;
* две стороны и угол между ними (угол задан в радианах);
* три стороны.

1. Составить программу, которая запрашивает номер группы элементов, вводит соответствующие элементы и вычисляет площадь треугольника. Вычисления прекратить, если в качестве номера группы введен 0.
2. Дано натуральное число *N*. Вычислить:

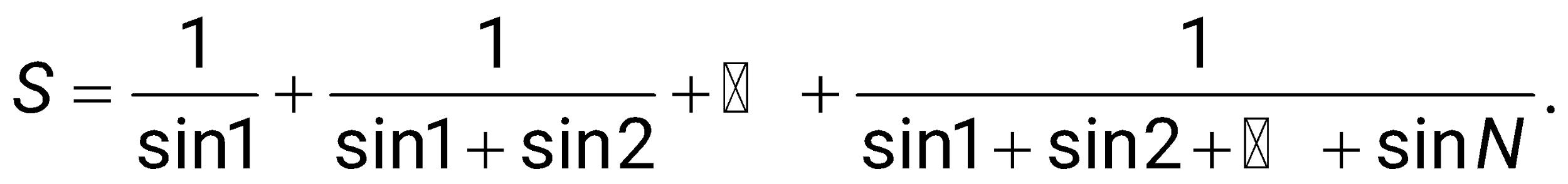
S=

1. Найти сумму ряда с точностью ε=0,001, общий член которого:

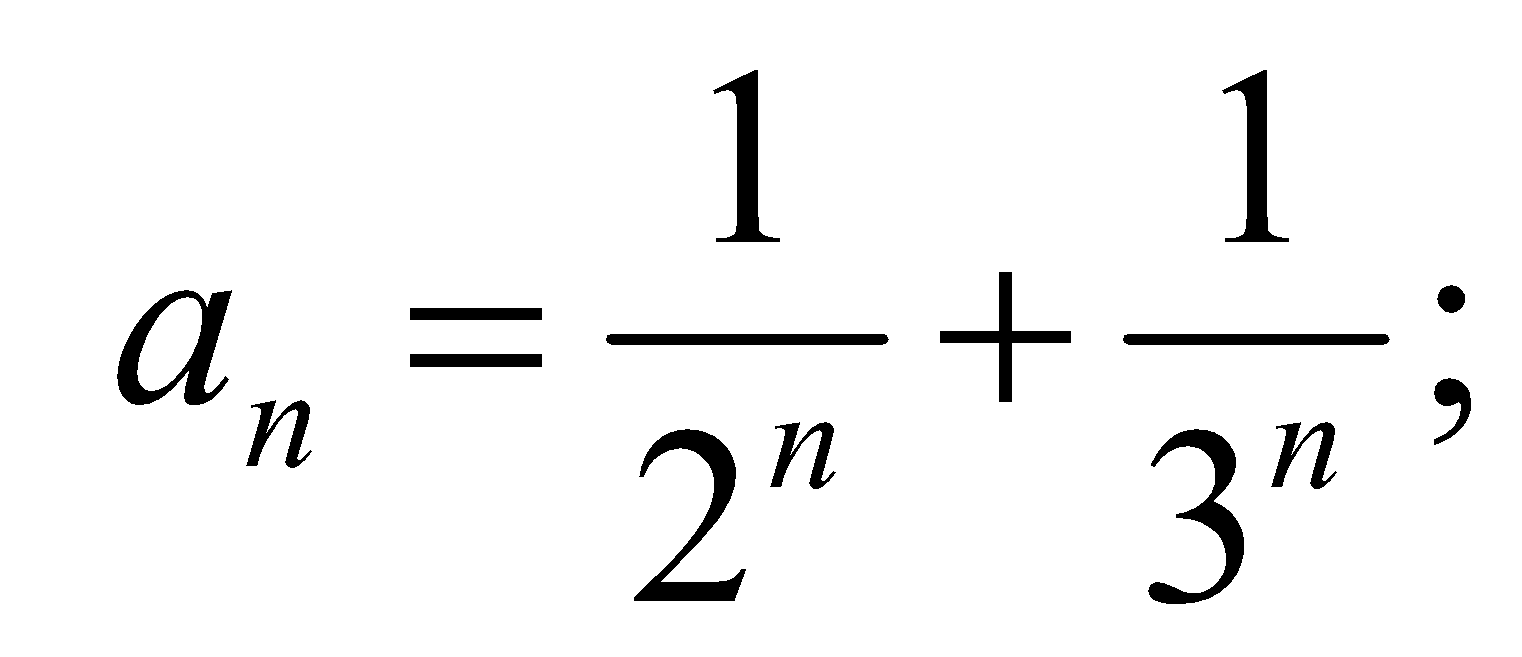
******

***Вариант 2.***

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 7 дней?
2. Дано натуральное число N. Вычислить:



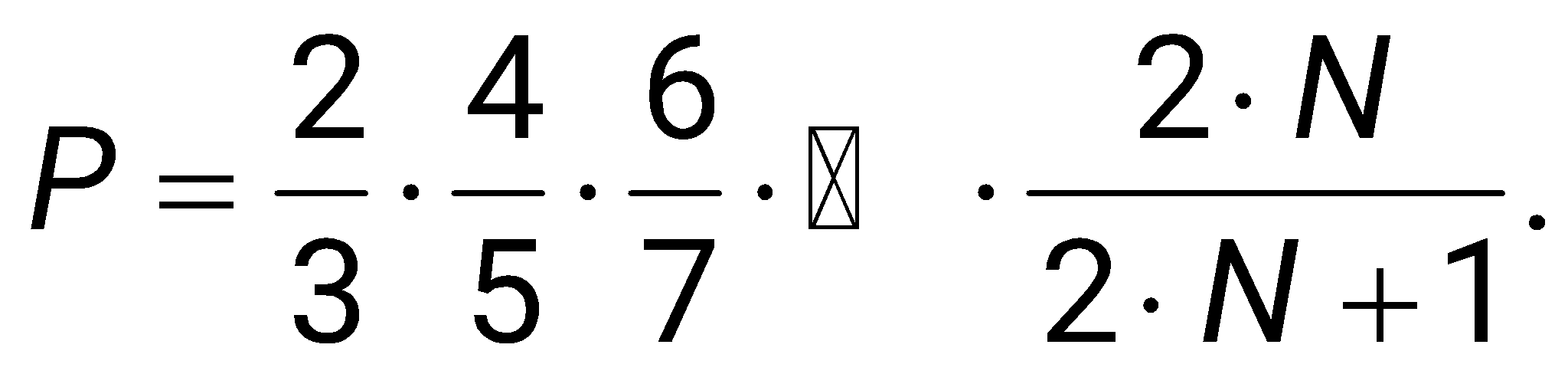
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



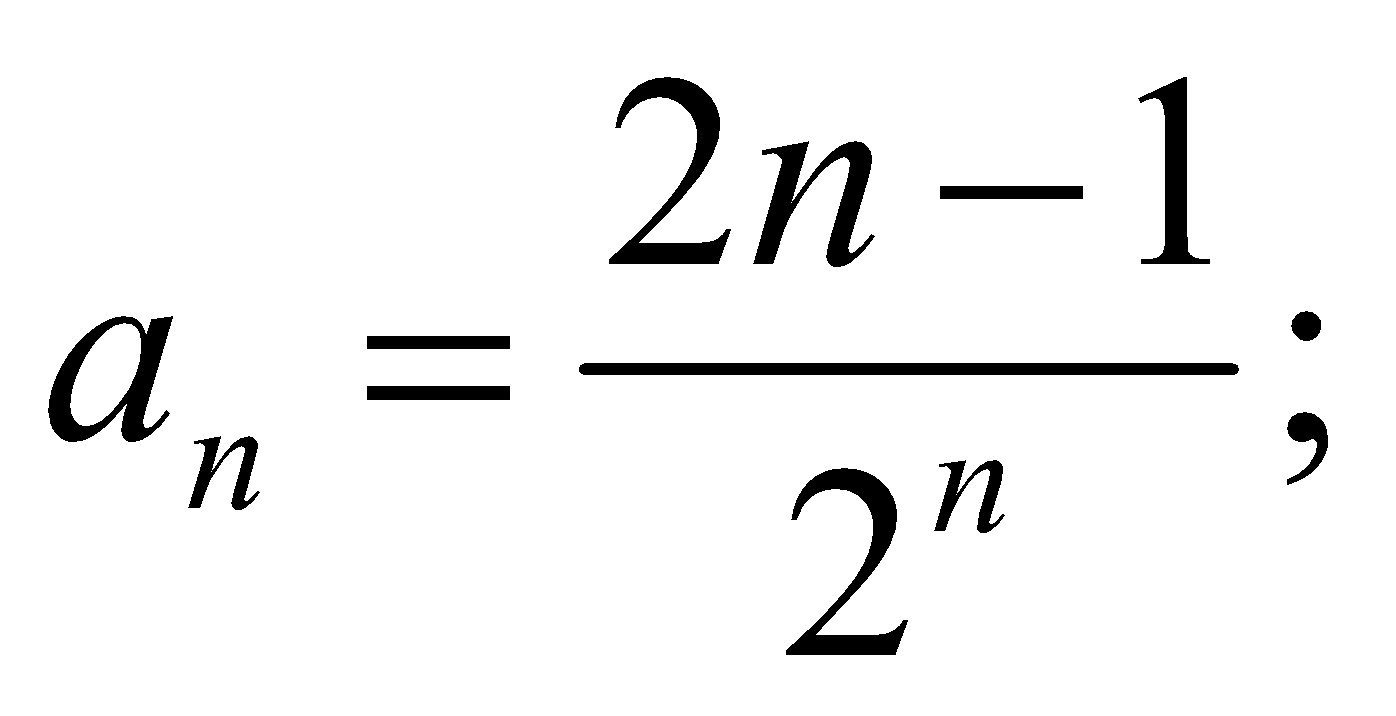
1. Дано целое число *N* и набор из *N* вещественных чисел. Вывести сумму и произведение чисел из данного набора.

***Вариант 3.***

1. Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько амеб будет через 3, 6, 9, 12, ... , 24 часа ?
2. Дано натуральное число *N*. Вычислить произведение первых *N* сомножителей:



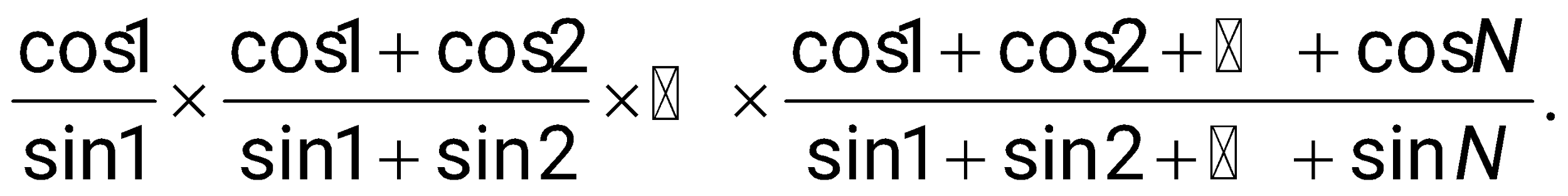
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



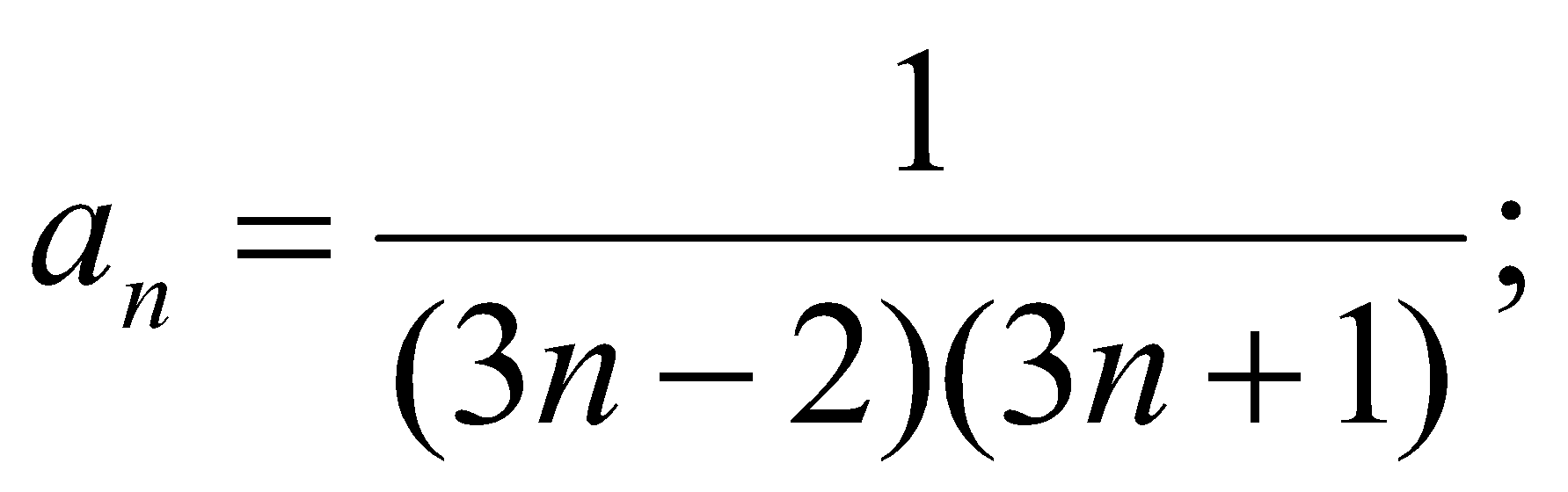
1. Даны десять вещественных чисел. Найти их среднее арифметическое.

***Вариант 4.***

1. Около стены наклонно стоит палка длиной х м. Один ее конец находится на расстоянии *y* м от стены. Определить значение угла α между палкой и полом для значений *x*=*k* м и *y,* изменяющегося от 2 до 3 м с шагом *h* м.
2. Дано натуральное число N. Вычислить:



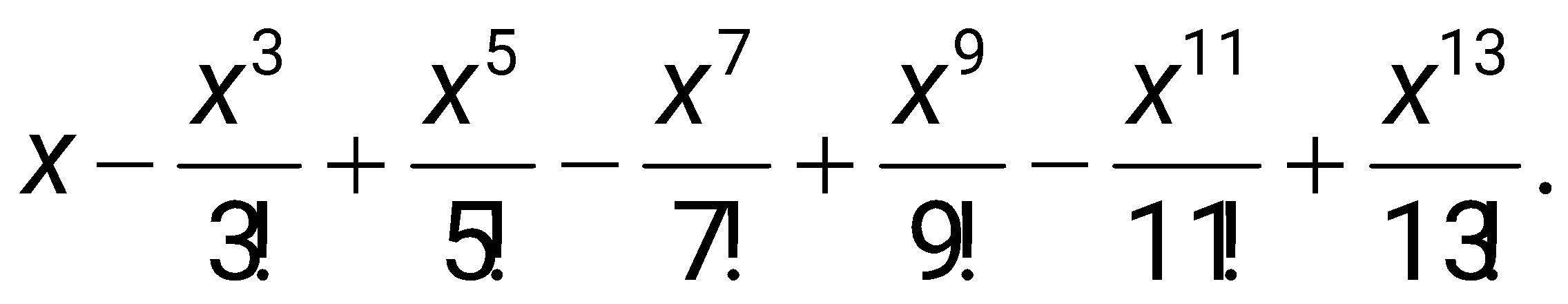
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



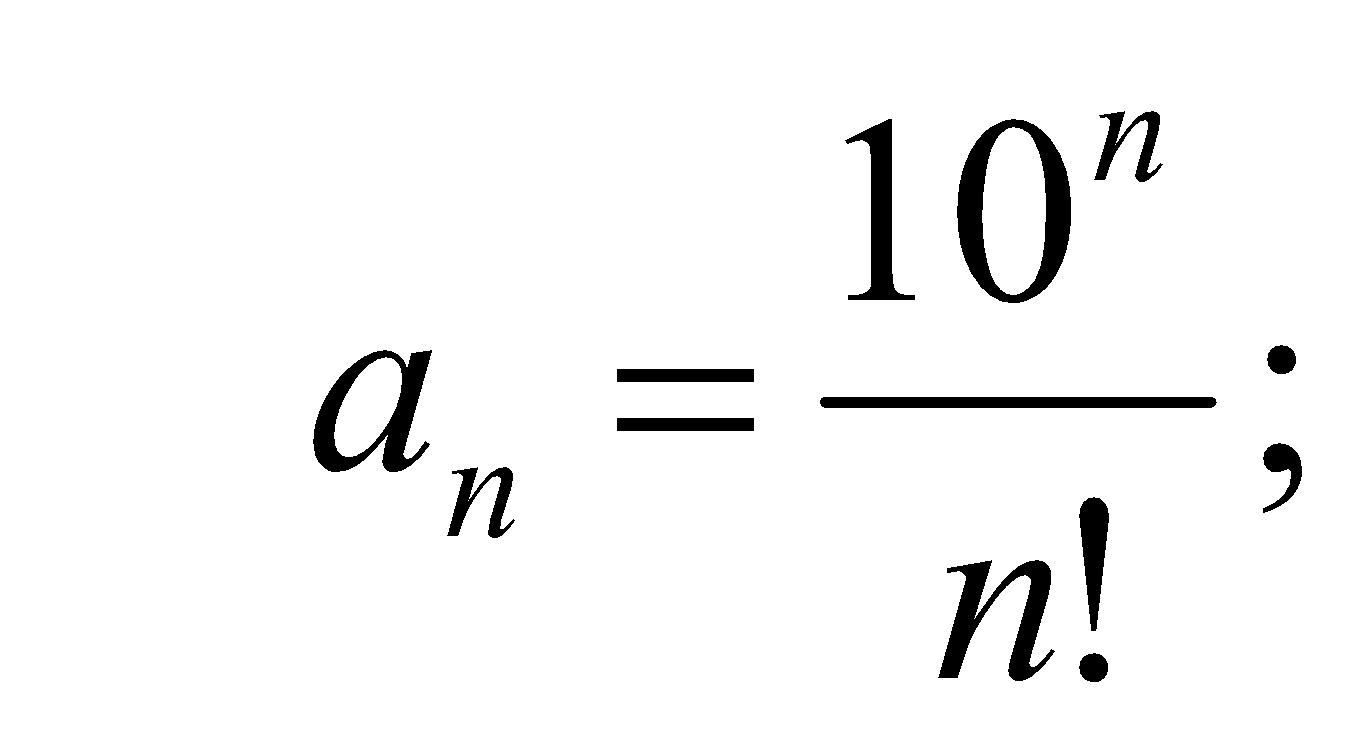
1. Дано целое число *N* и набор из *N* положительных вещественных чисел. Вывести в том же порядке целые части всех чисел из данного набора (как вещественные числа с нулевой дробной частью), а также сумму всех целых частей.

***Вариант 5.***

1. У гусей и кроликов вместе 64 лапы. Сколько могло быть кроликов и гусей (указать все сочетания, которые возможны)?
2. Дано действительное число. Вычислить:



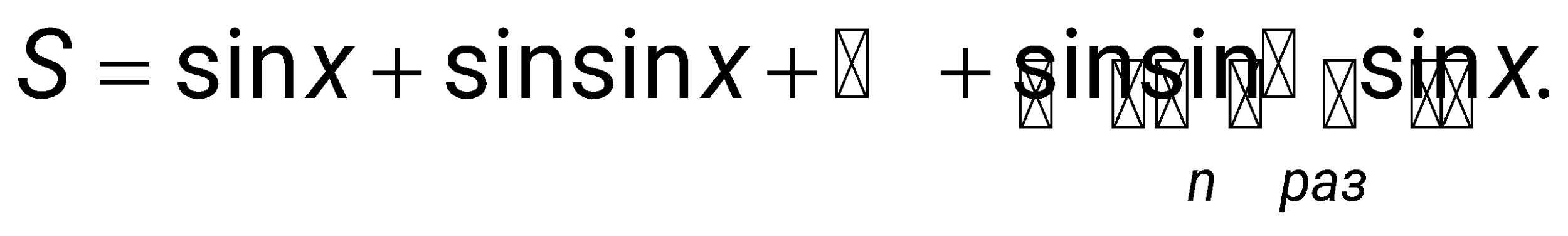
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



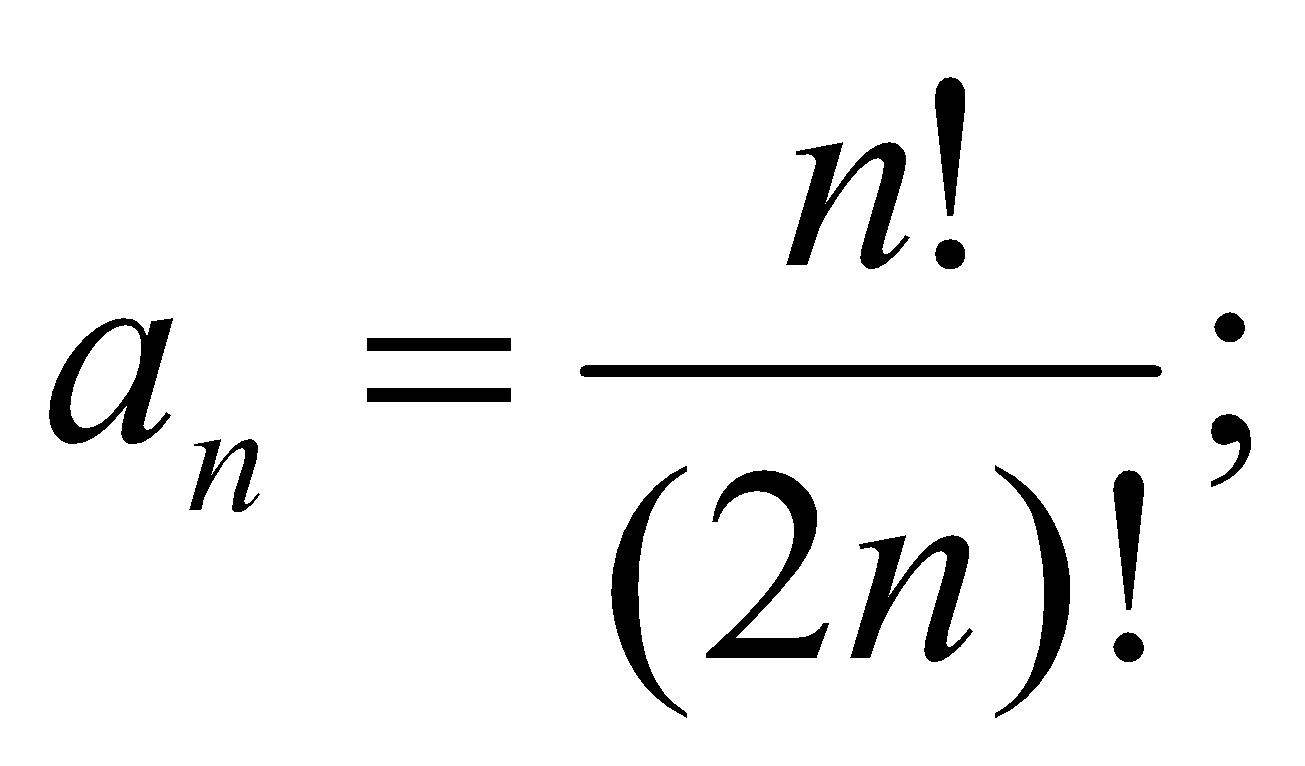
1. Дано целое число *N* и набор из *N* положительных вещественных чисел. Вывести в том же порядке дробные части всех чисел из данного набора (как вещественные числа с нулевой целой частью), а также произведение всех дробных частей.

***Вариант 6.***

1. Составить алгоритм решения задачи: сколько можно купить быков, коров и телят, платя за быка 10 р., за корову - 5 р., а за теленка - 0,5 р., если на 100 р. надо купить 100 голов скота?
2. Даны: натуральное *n*, действительное *х*. Вычислить:



1. Найти сумму ряда с точностью ε=0,001, общий член которого:



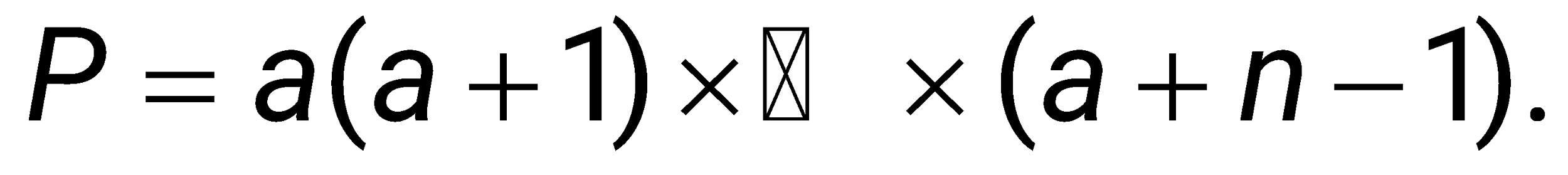
1. Дано целое число *N* и набор из *N* вещественных чисел. Вывести в том же порядке округленные значения всех чисел из данного набора (как целые числа), а также сумму всех округленных значений.

***Вариант 7.***

1. Доказать (путем перебора возможных значений), что для любых величин А, B, С типа Boolean следующие пары логических выражений имеют одинаковые значения (эквивалентны):

* A OR B и B OR A;
* A AND и B AND ;
* (A OR В) OR С и A OR С;
* (A AND В) AND С и A AND (В AND С);
* A AND (A OR В) и А;
* A OR (A AND В) и А;
* A AND (В OR С) и (A AND В) OR (A AND С);
* A OR (В AND С) и (A OR В) AND (A OR С).

1. Даны: действительное число *a*, натуральное число *n*. Вычислить:



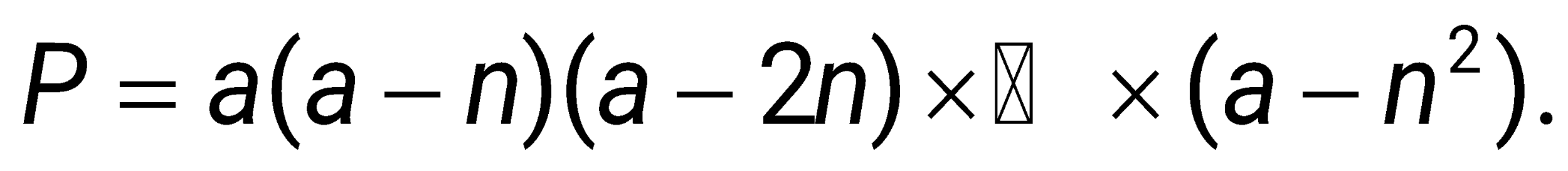
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



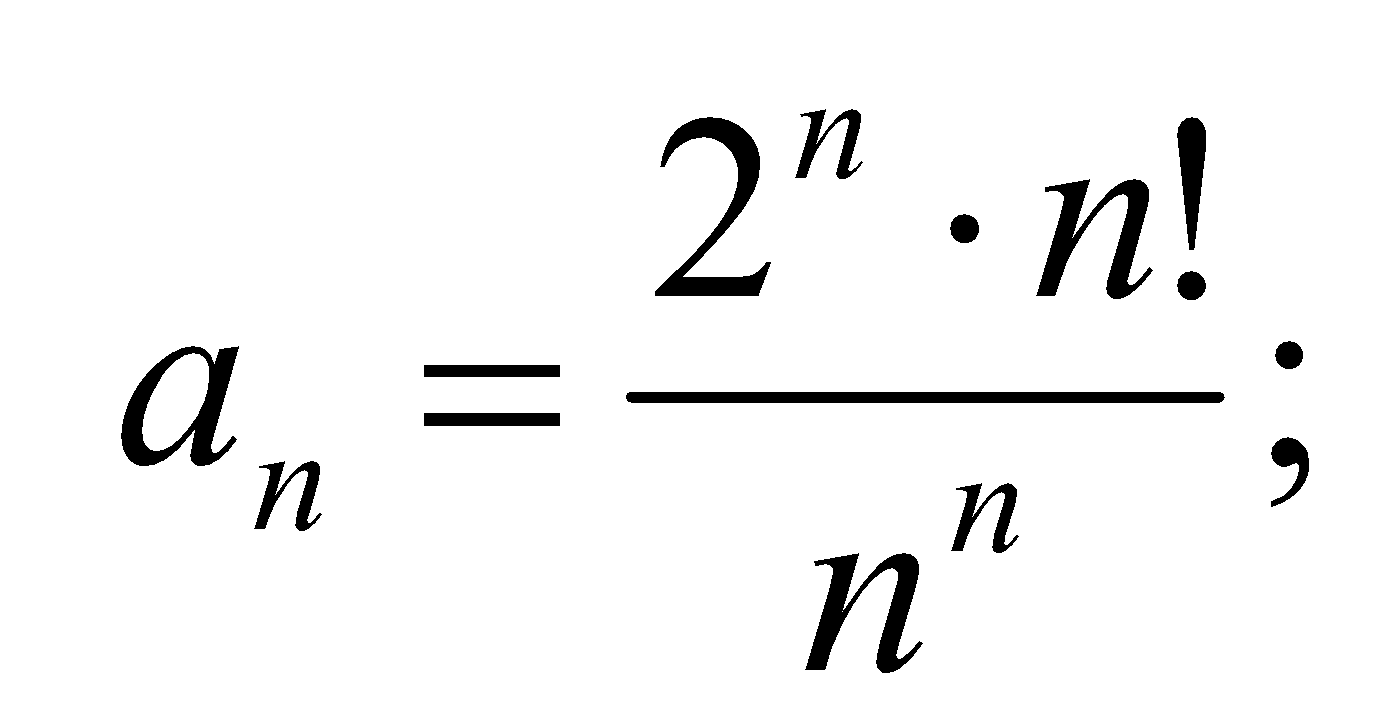
1. Дано целое число *N* и набор из *N* целых чисел. Вывести в том же порядке все четные числа из данного набора и количество *K* таких чисел.

***Вариант 8.***

1. Составить программу для проверки утверждения: «Результатами вычислений по формуле *х*2+*х*+17 при 0≤*х≤*15 являются простые числа». Все результаты вывести на экран.
2. Даны: действительное число *a*, натуральное число *n*. Вычислить:



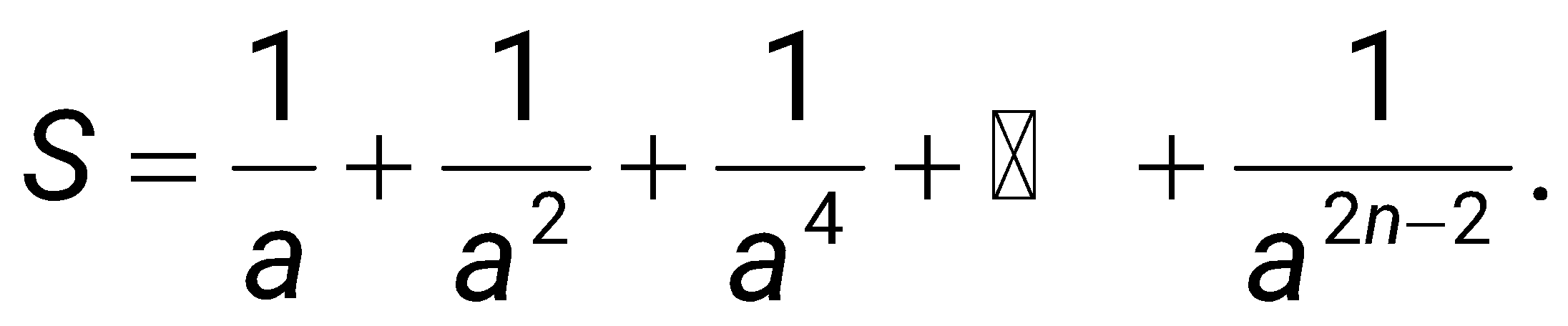
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



1. Дано целое число *N* и набор из *N* целых чисел. Вывести в том же порядке номера всех нечетных чисел из данного набора и количество *K* таких чисел.

***Вариант 9.***

1. Составить программу для проверки утверждения: «Результатами вычислений по формуле *х*2+*х*+41 при 0≤*х≤*40 являются простые числа.». Все результаты вывести на экран.
2. Даны: действительное число *а*, натуральное число *п*. Вычислить:



1. Найти сумму ряда с точностью ε=10-3, общий член которого:



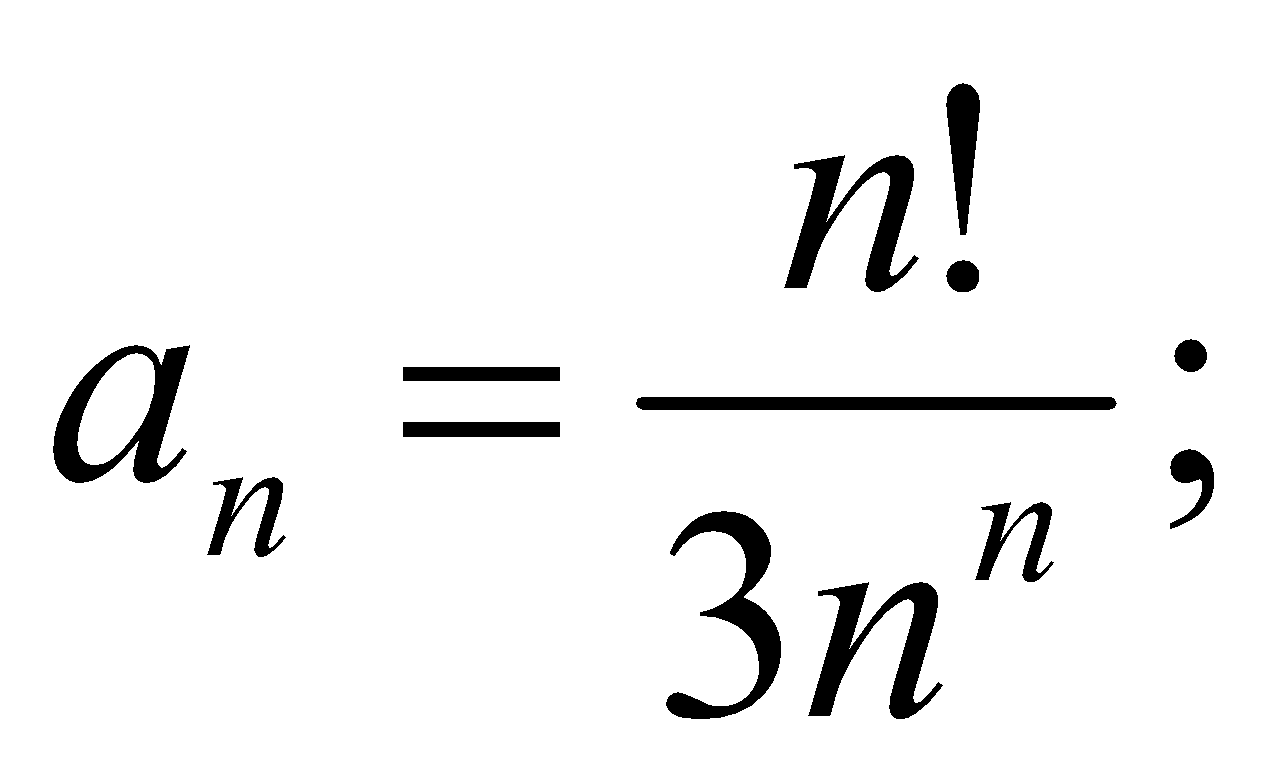
1. Дано целое число *N* и набор из *N* целых чисел. Если в наборе имеются положительные числа, то вывести True; в противном случае вывести False.

***Вариант 10.***

1. Составить программу-генератор простых чисел, в основу положить формулу 2*х*2+29 при 0≤*х≤*28.
2. Дано действительное *x*. Вычислить:

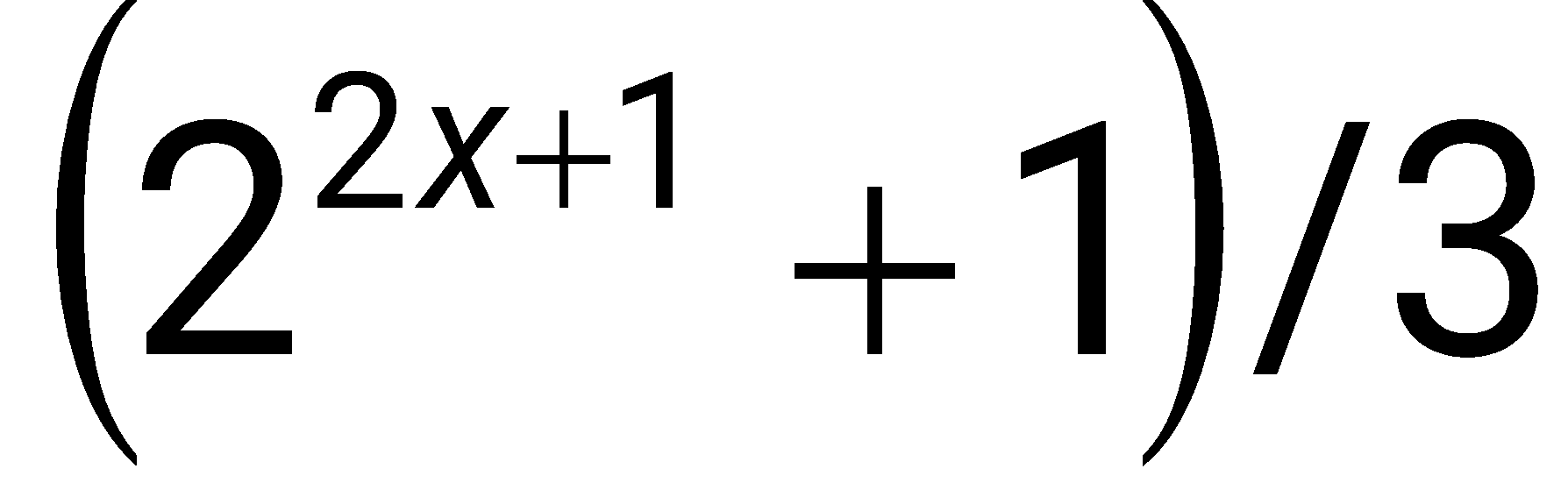


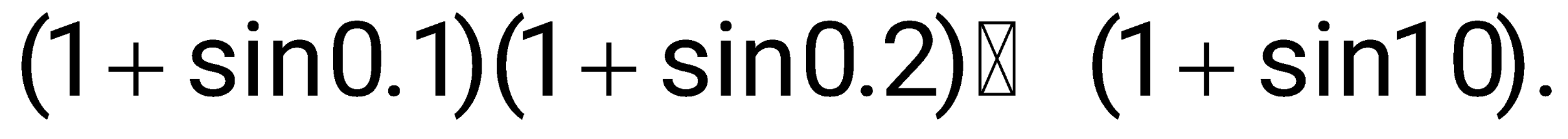
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



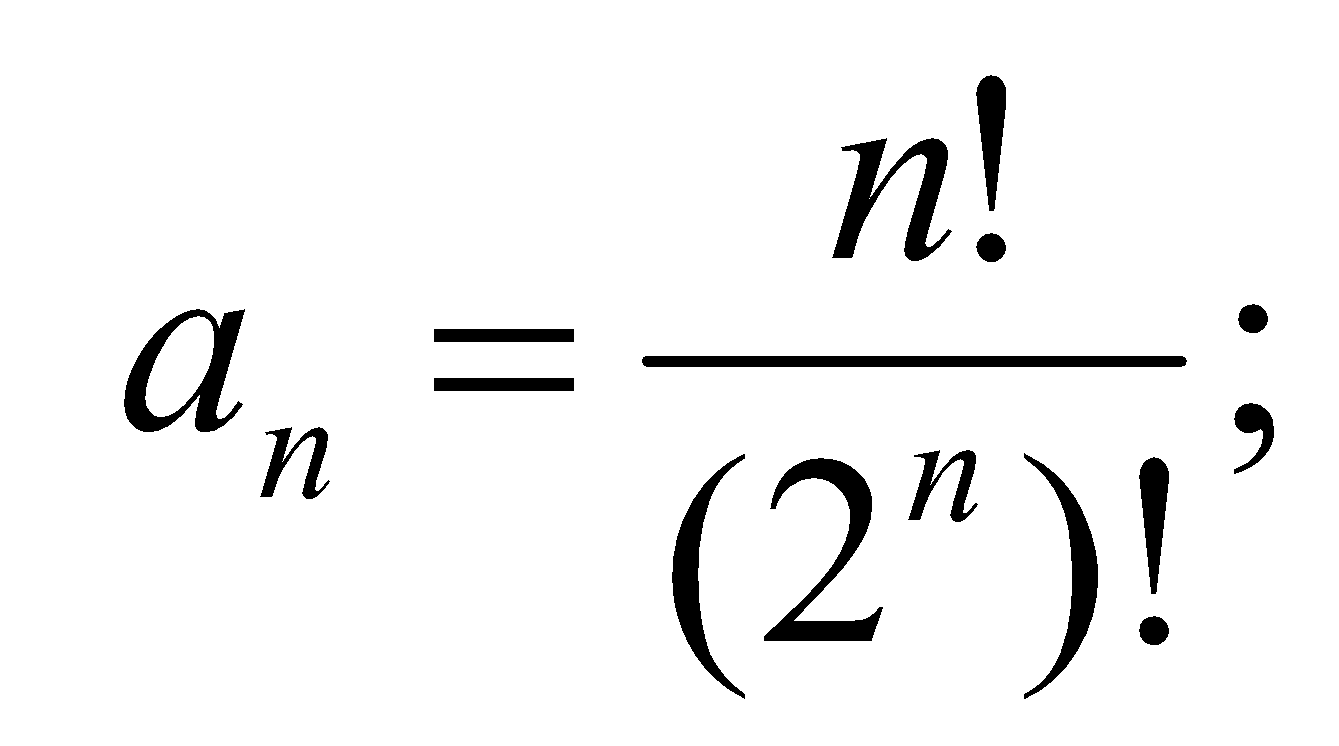
1. Дан набор ненулевых целых чисел; признак его завершения — число 0. Вывести количество чисел в наборе.

***Вариант 11.***

1. Составить программу-генератор простых чисел, в основу положить формулу при 1≤*х≤*Зб.
2. Вычислить:

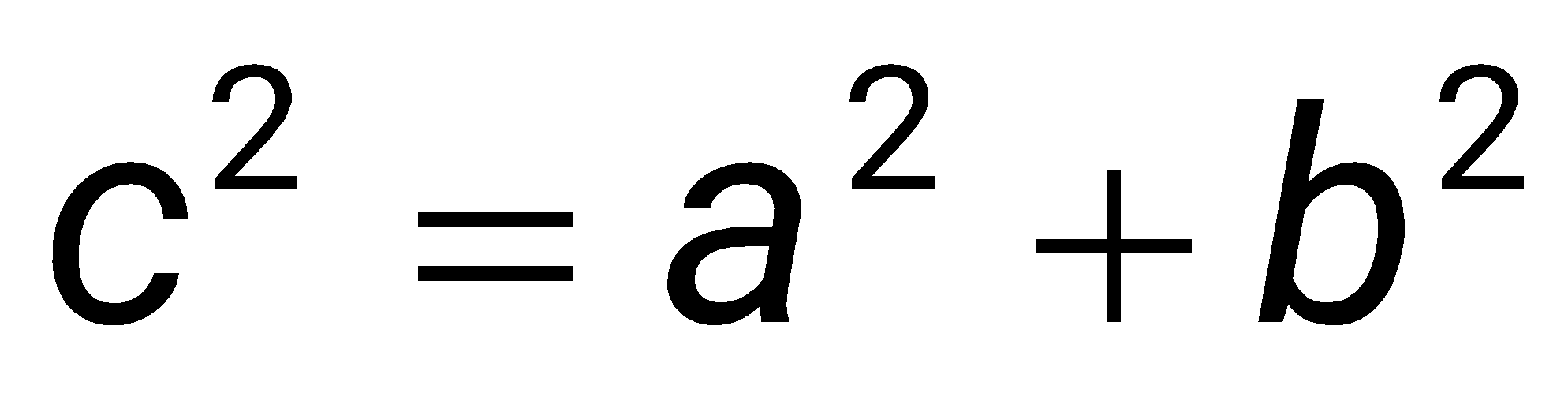
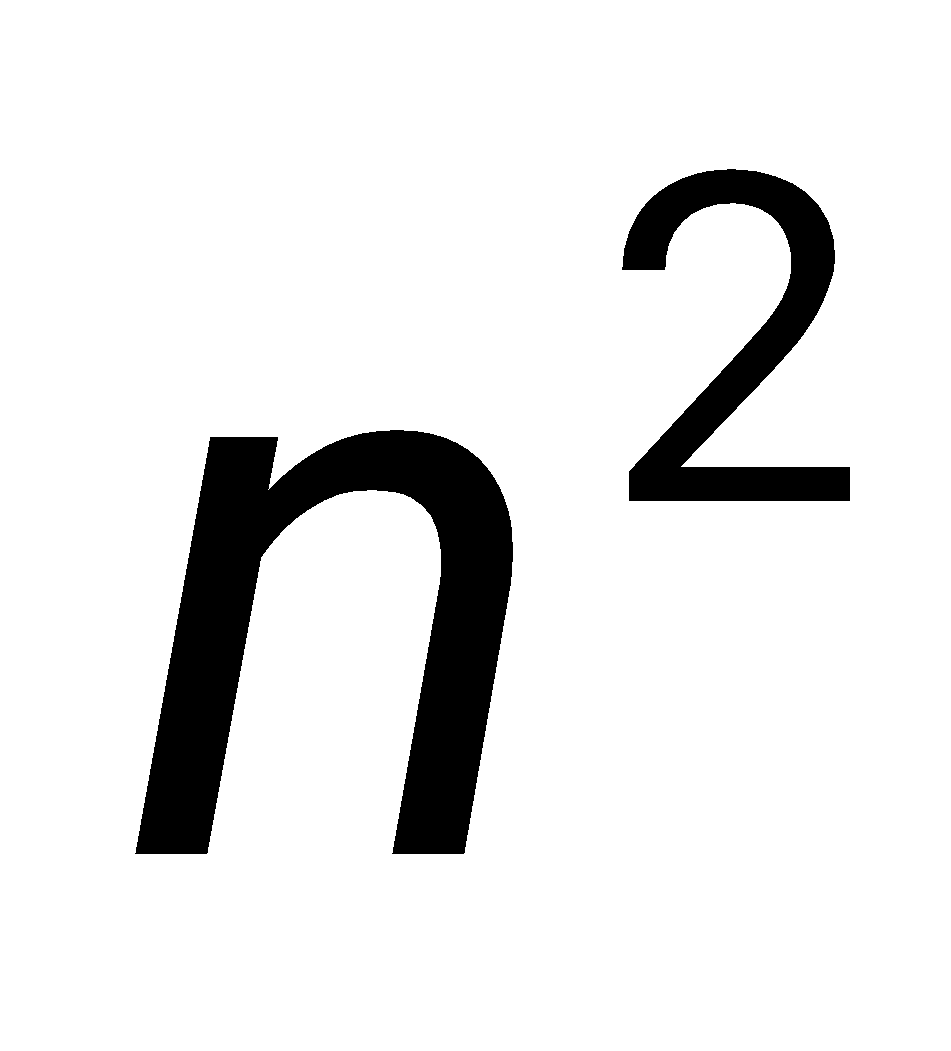
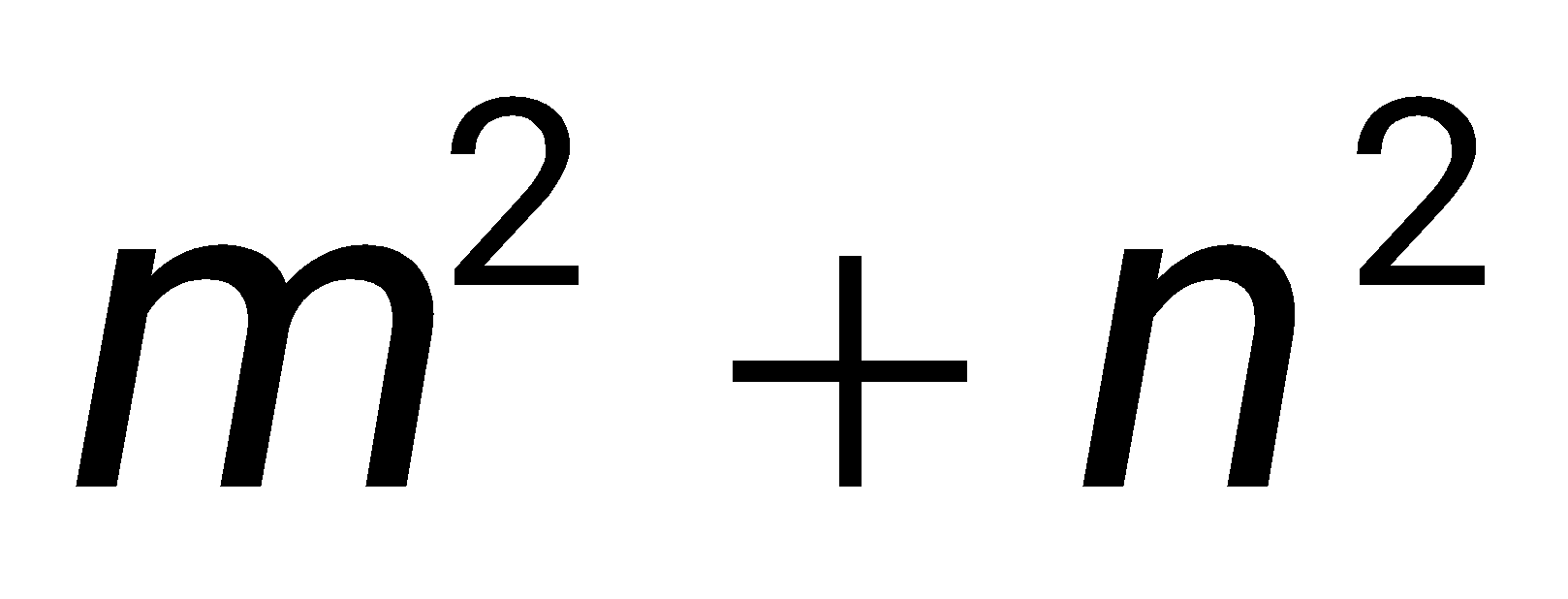


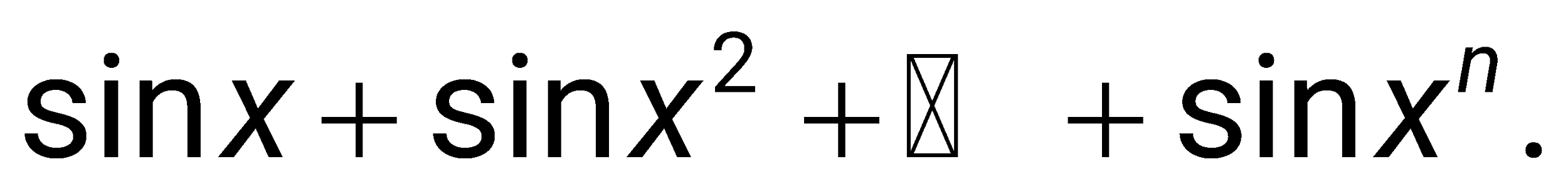
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



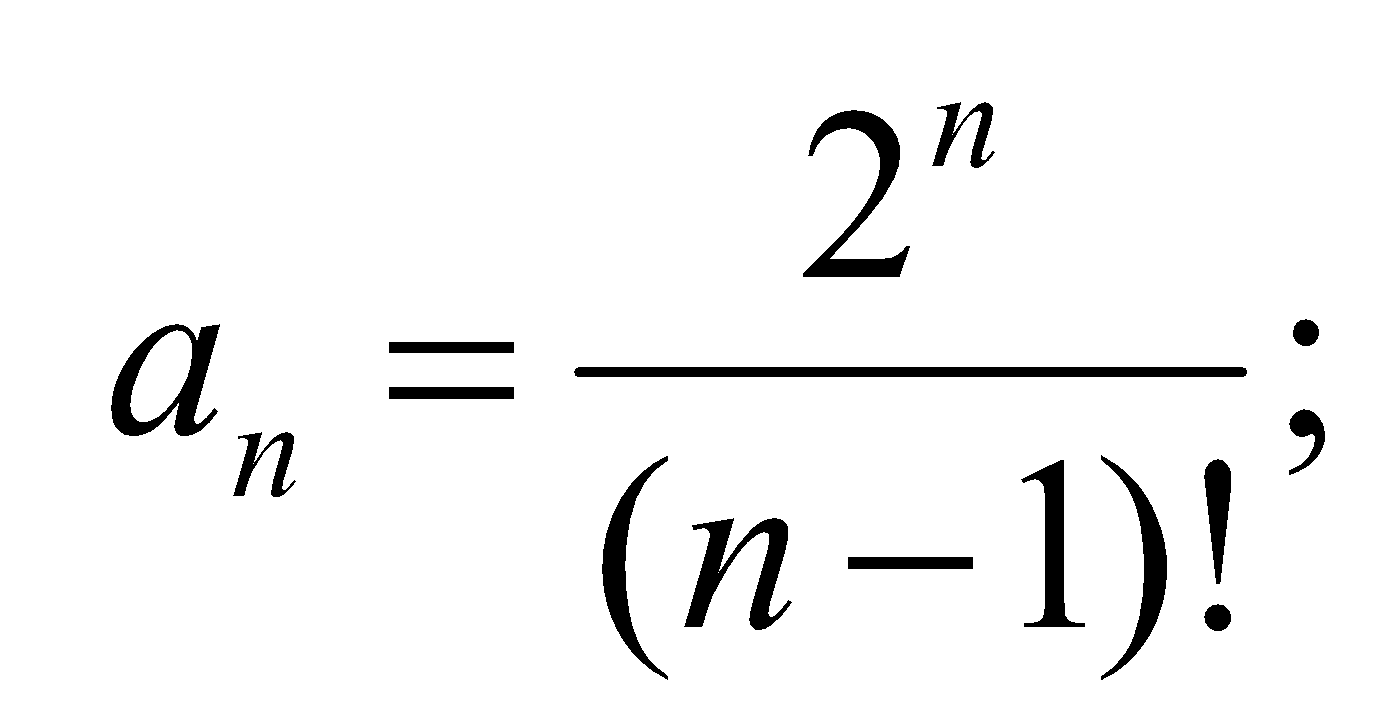
1. Дан набор ненулевых целых чисел; признак его завершения — число 0. Вывести сумму всех положительных четных чисел из данного набора. Если требуемые числа в наборе отсутствуют, то вывести 0.

***Вариант 12.***

1. Составить программу-генератор чисел Пифагора *а, b, с, (*).В основу положить формулы: *а*=*т*2-,*b*=2*тn,* с=(*m, n-*натуральные, 1<*m*<*k*, 1<*n*<*k, k* - данное число). Результат вывести на; экран в виде таблицы из пяти столбцов: *m, n, a, b, c.*
2. Даны натуральное *п*, действительное *х*. Вычислить:



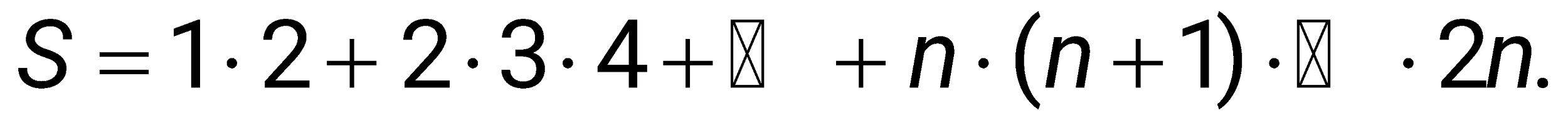
1. Найти сумму ряда с точностью ε=10-3, общий член которого:

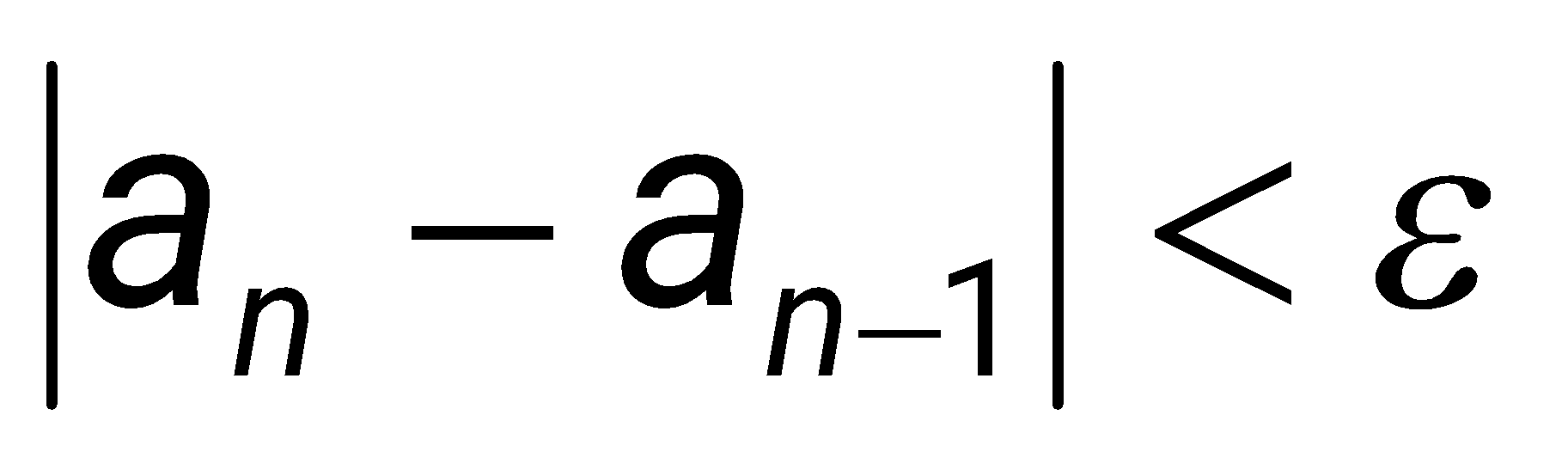
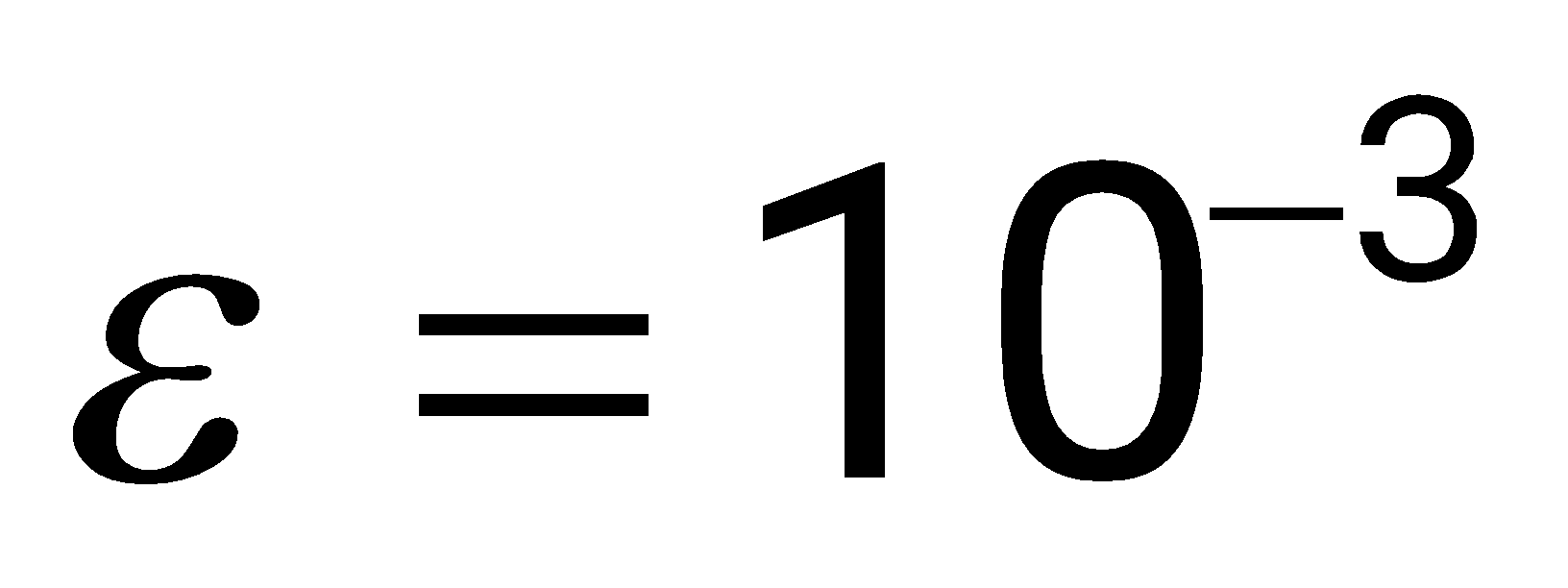
******

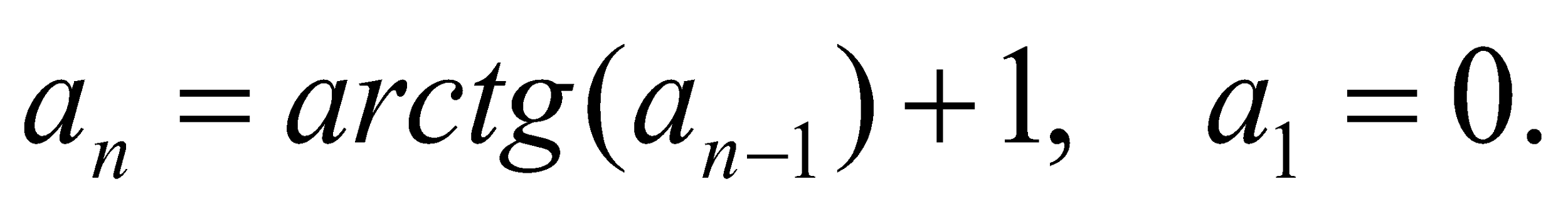
1. Дано целое число *N* и набор из *N* целых чисел. Если в наборе имеются положительные числа, то вывести True; в противном случае вывести False.

***Вариант 13.***

1. Покупатель должен заплатить в кассу S р.У него имеются 1, 5, ,10, 50, 100, 500, 1000 и 10000р.Сколько купюр разного достоинства отдаст покупатель, если он начинает платить с самых крупных купюр?
2. Дано натуральное *п*. Вычислить:



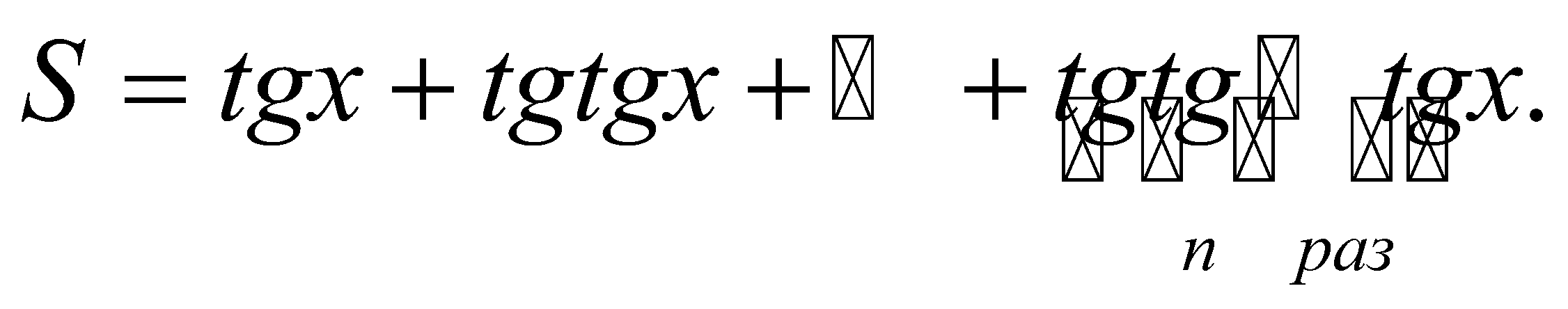
1. Найти наименьший номер последовательности, для которого выполняется условие . Вывести на экран этот номер и все элементы *ai*, где *i*=1,2,...,*п*, если 



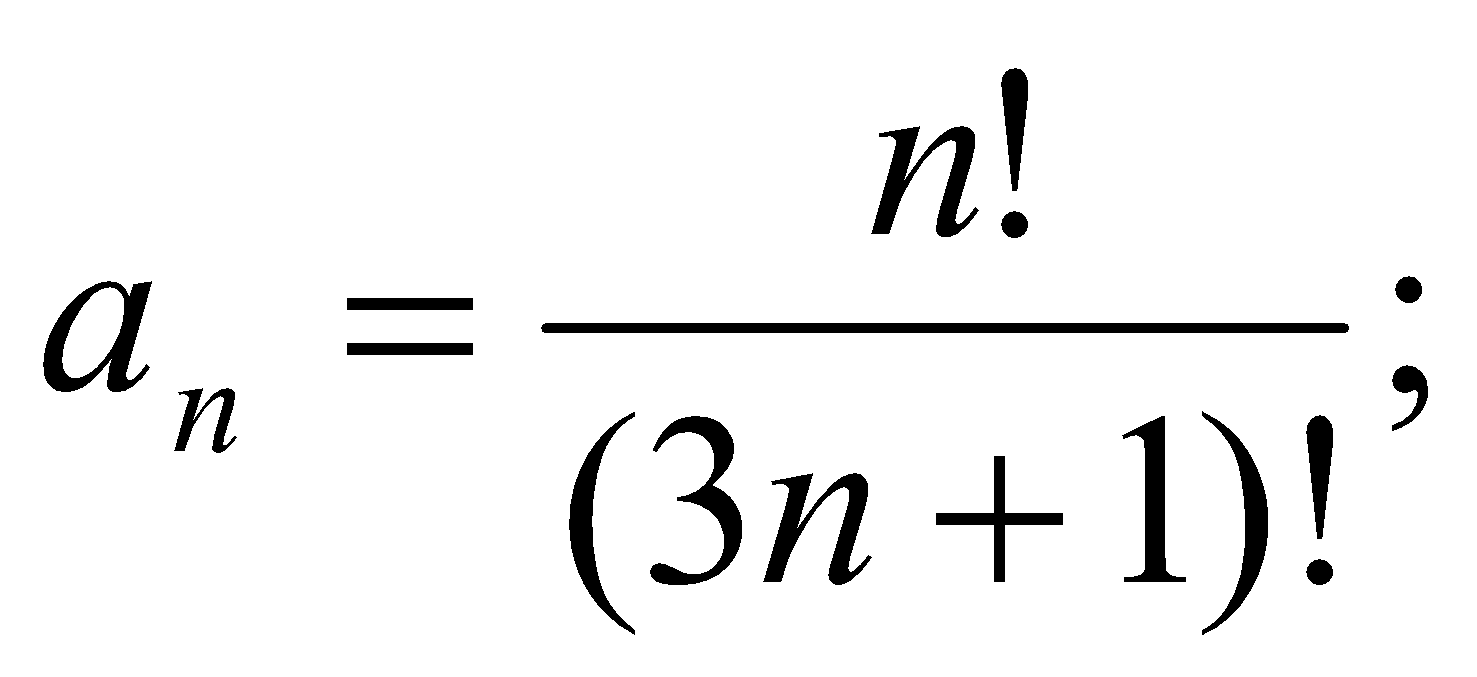
1. Дано целое число *K* и набор ненулевых целых чисел; признак его завершения — число 0. Вывести количество чисел в наборе, меньших *K*.

***Вариант 14.***

1. Найти наибольший общий делитель трех заданных натуральных чисел,  
   используя алгоритм Евклида и учитывая, что НОД(*a*, *b*, *c*) = НОД(НОД(*a*, *b*), *c*).
2. Даны: натуральное *n*, действительное *х*. Вычислить:



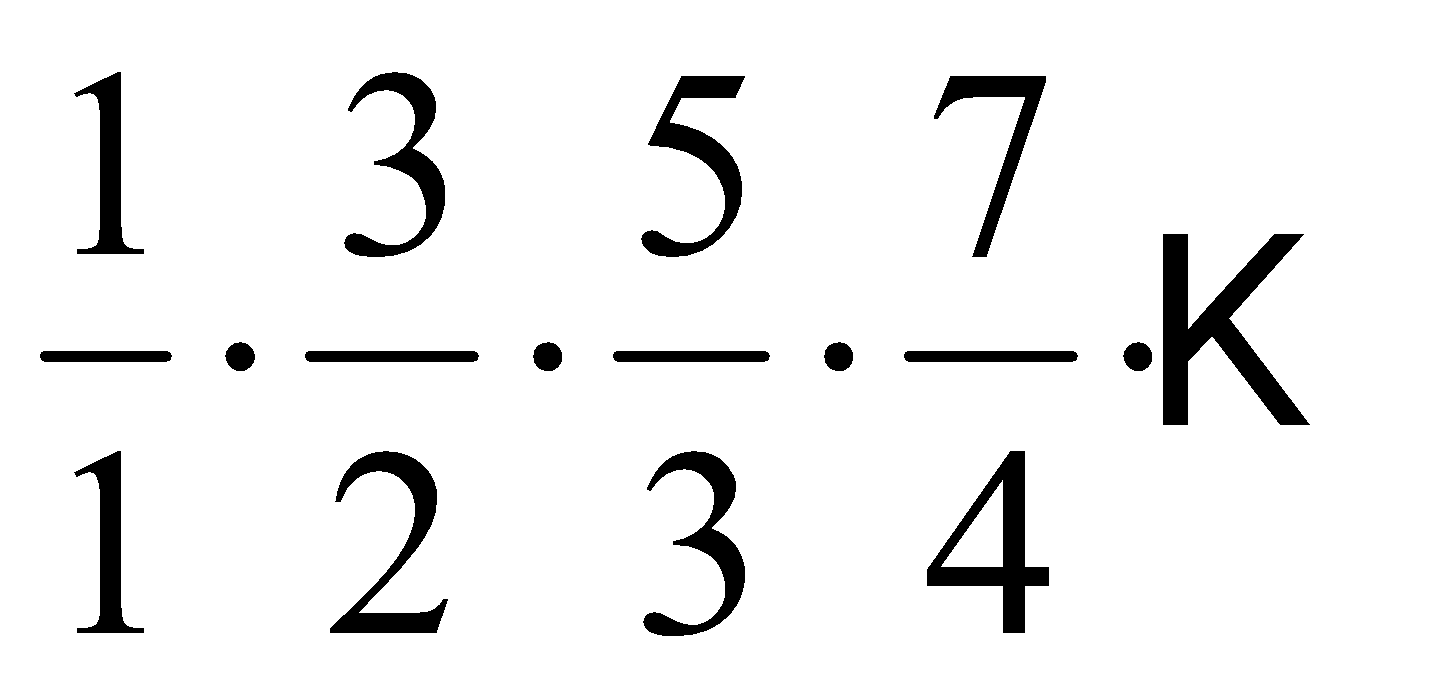
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



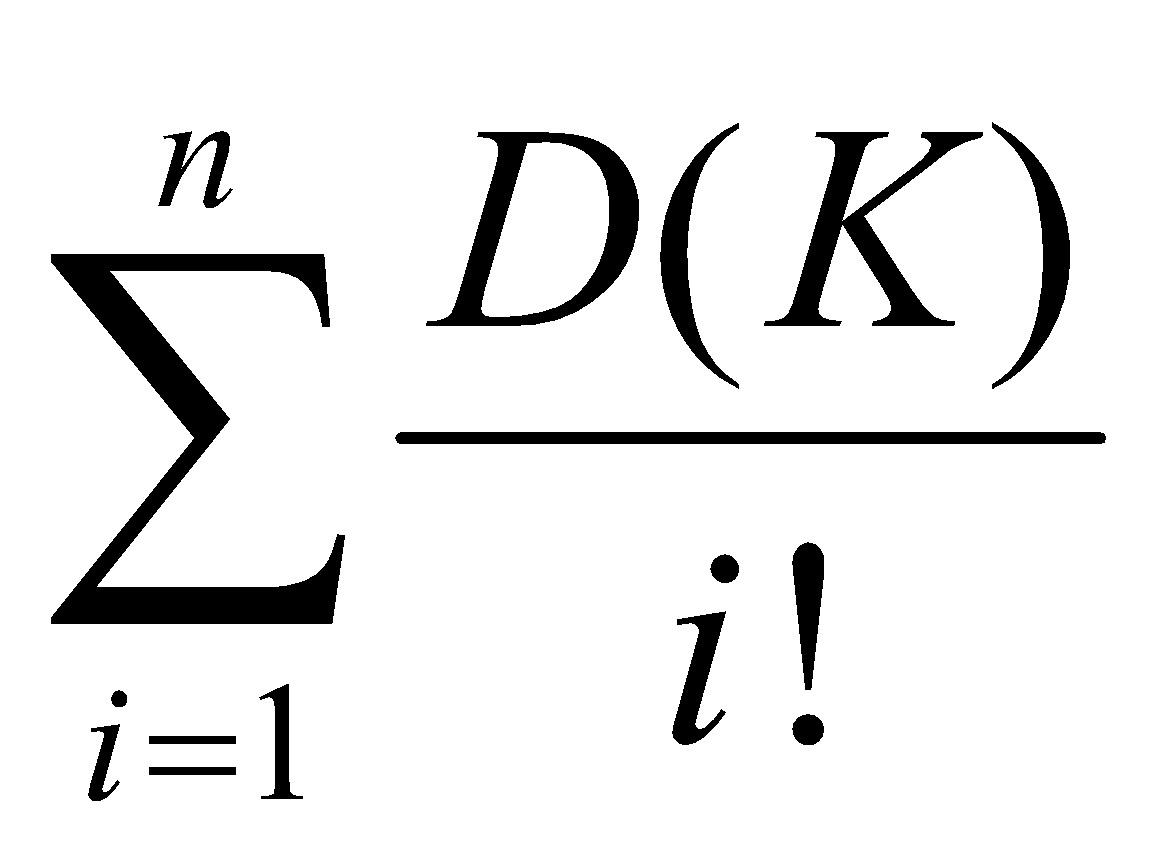
1. Дано целое число *K*, а также *K* наборов ненулевых целых чисел. Каждый набор содержит не менее трех элементов, признаком его завершения является число 0. Для каждого набора выполнить следующее действие: если набор является пилообразным, то вывести количество его элементов; в противном случае вывести номер первого элемента, который не является зубцом.

***Вариант 15.***

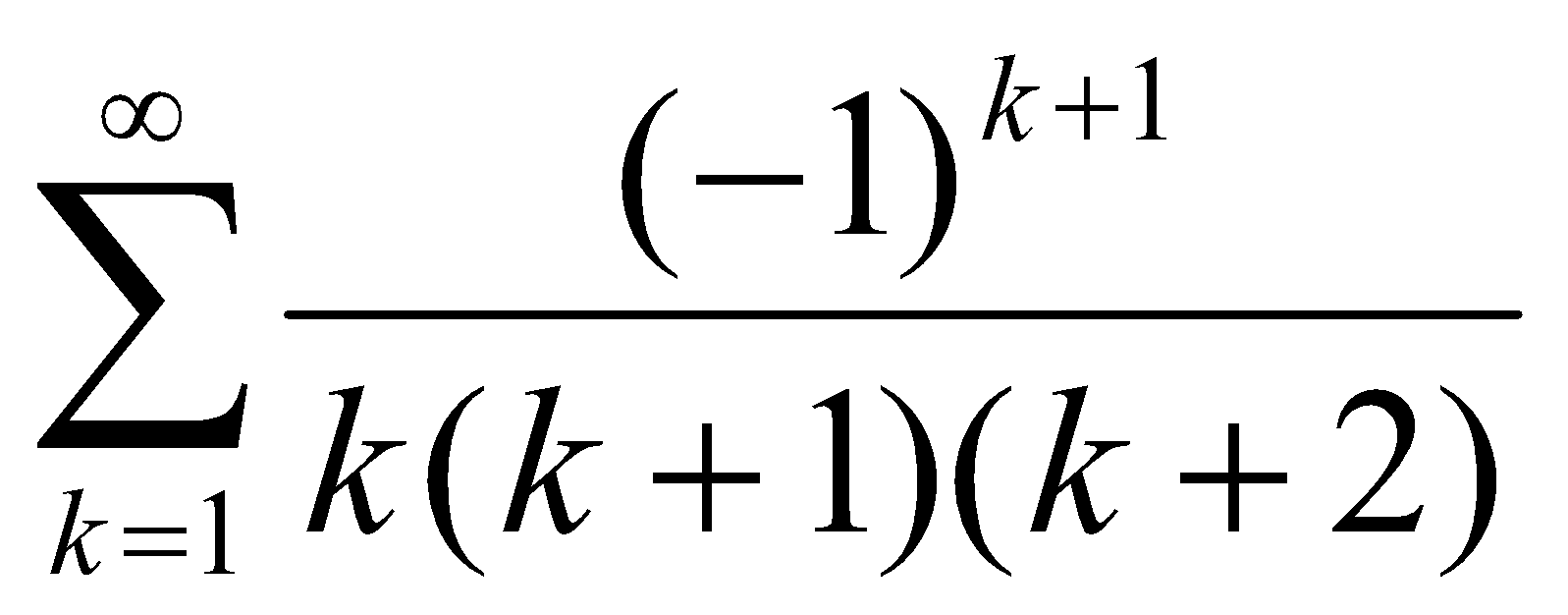
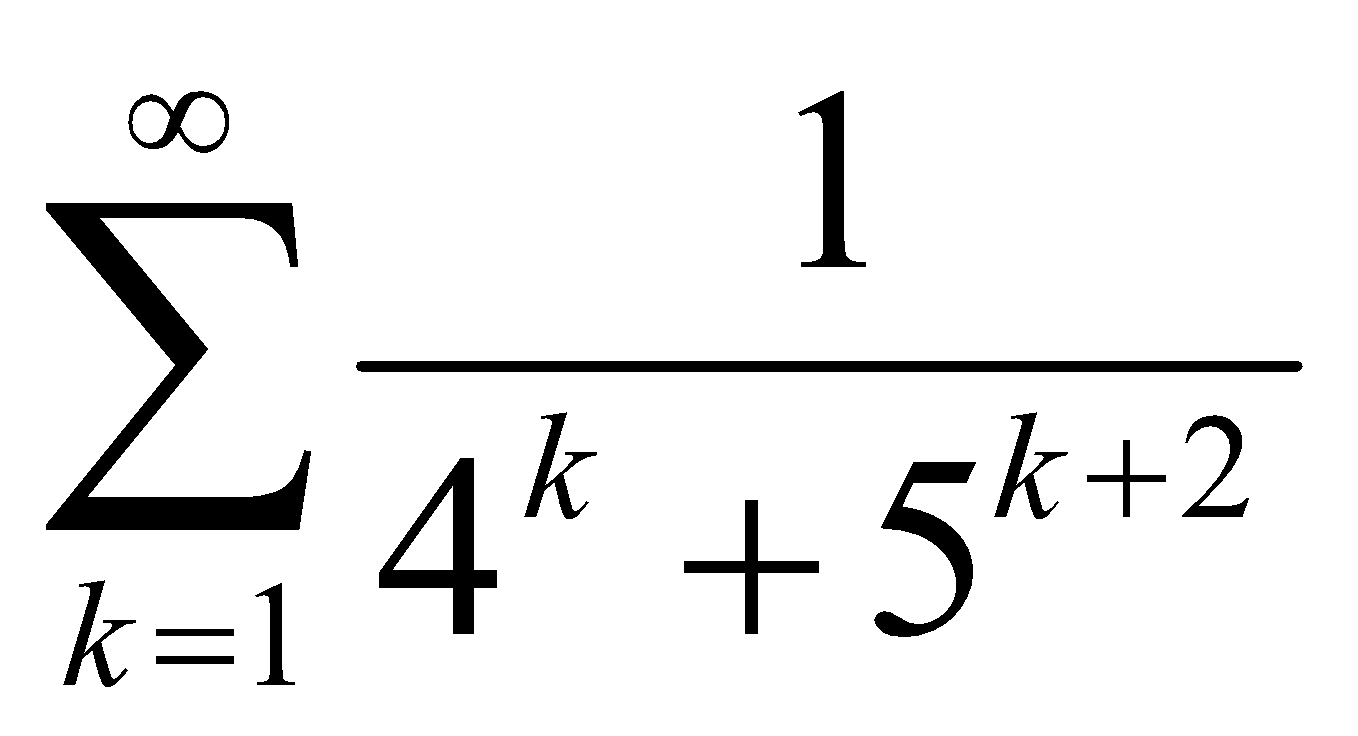
1. Дано натуральное число n. Вычислить произведение первых n сомножителей



1. Для любого целого *К* обозначим количество цифр в его десятичной записи D(***К***). Дано натуральное число n**.** Вычислить



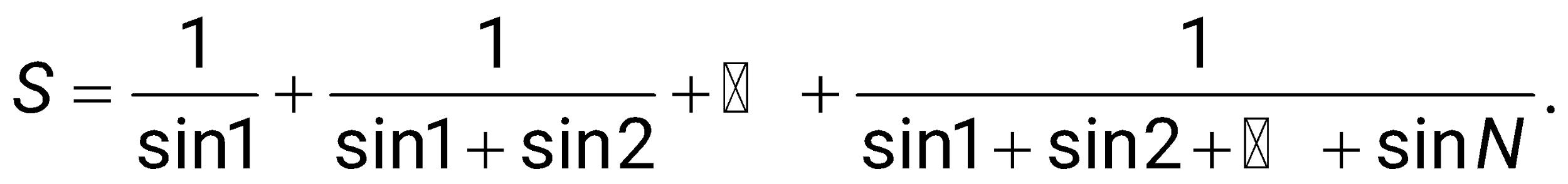
1. Вычислить бесконечную сумму с заданной точностью Eps (Eps>0). Требуемая точность достигнута, если очередное слагаемое оказалось по модулю меньше Eps. Вычислить

a)  b) 

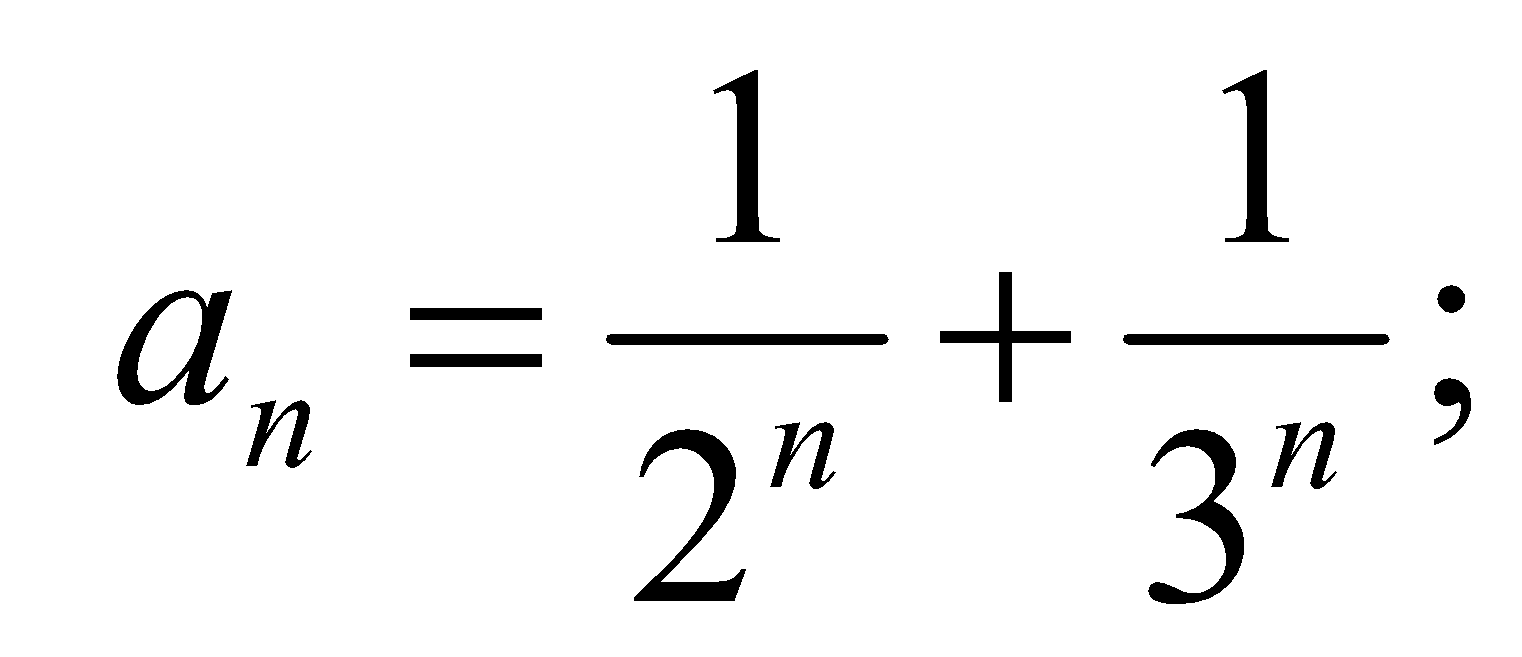
1. Дано целое число *K* и набор ненулевых целых чисел; признак его завершения — число 0. Вывести номер первого числа в наборе, большего *K*. Если таких чисел нет, то вывести 0.

***Вариант 16.***

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 7 дней?
2. Дано натуральное число N. Вычислить:



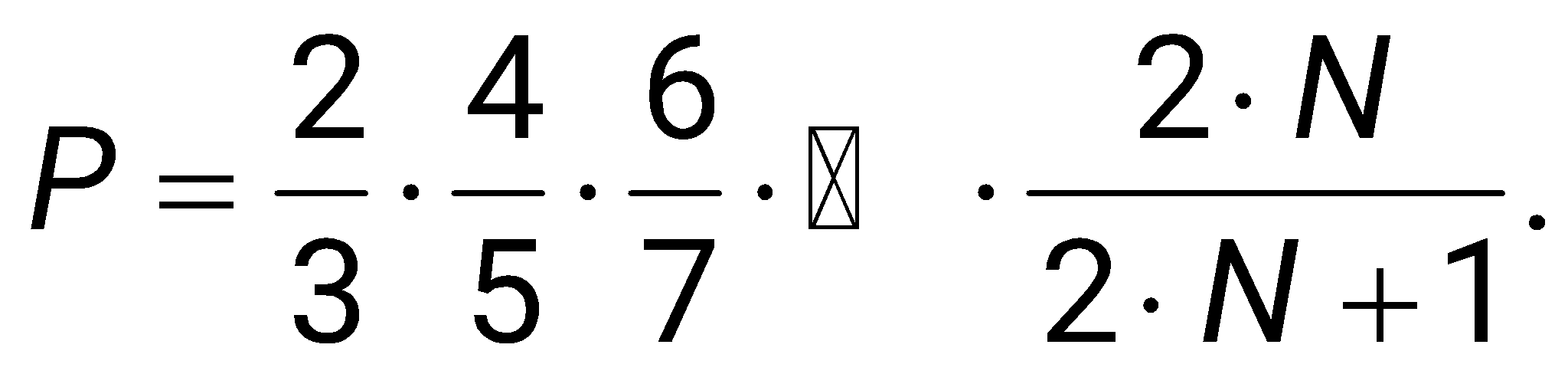
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



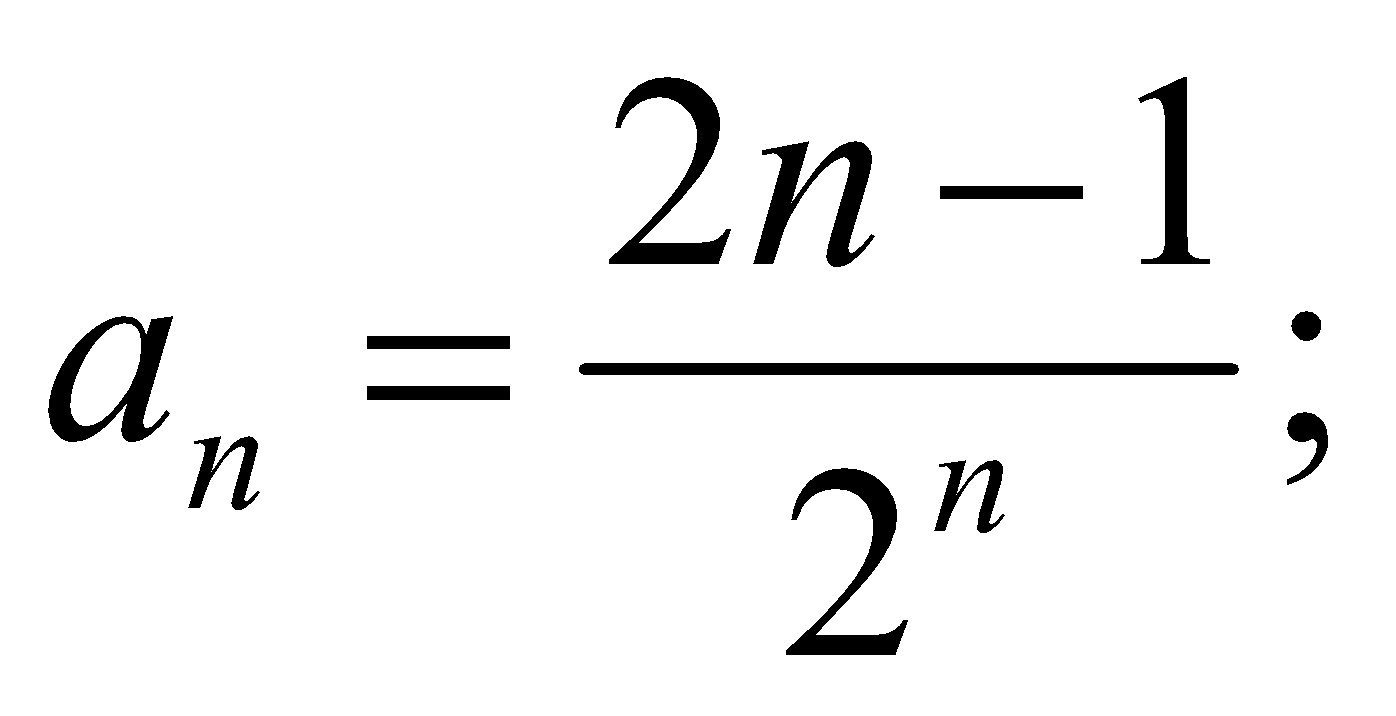
1. Дано целое число *K* и набор ненулевых целых чисел; признак его завершения — число 0. Вывести номер последнего числа в наборе, большего *K*. Если таких чисел нет, то вывести 0.

***Вариант 17.***

1. Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько амеб будет через 3, 6, 9, 12, ... , 24 часа ?
2. Дано натуральное число *N*. Вычислить произведение первых *N* сомножителей:



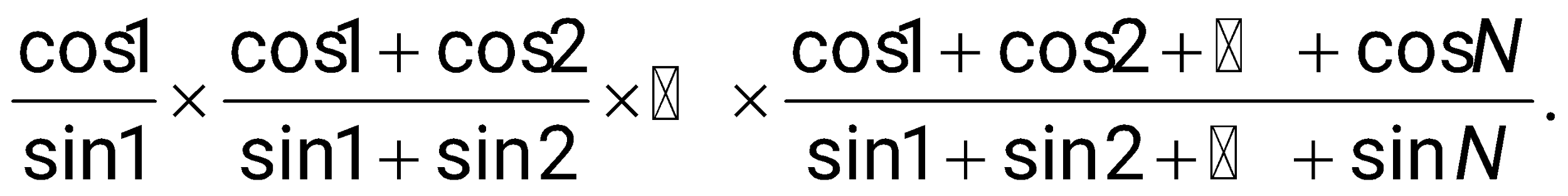
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



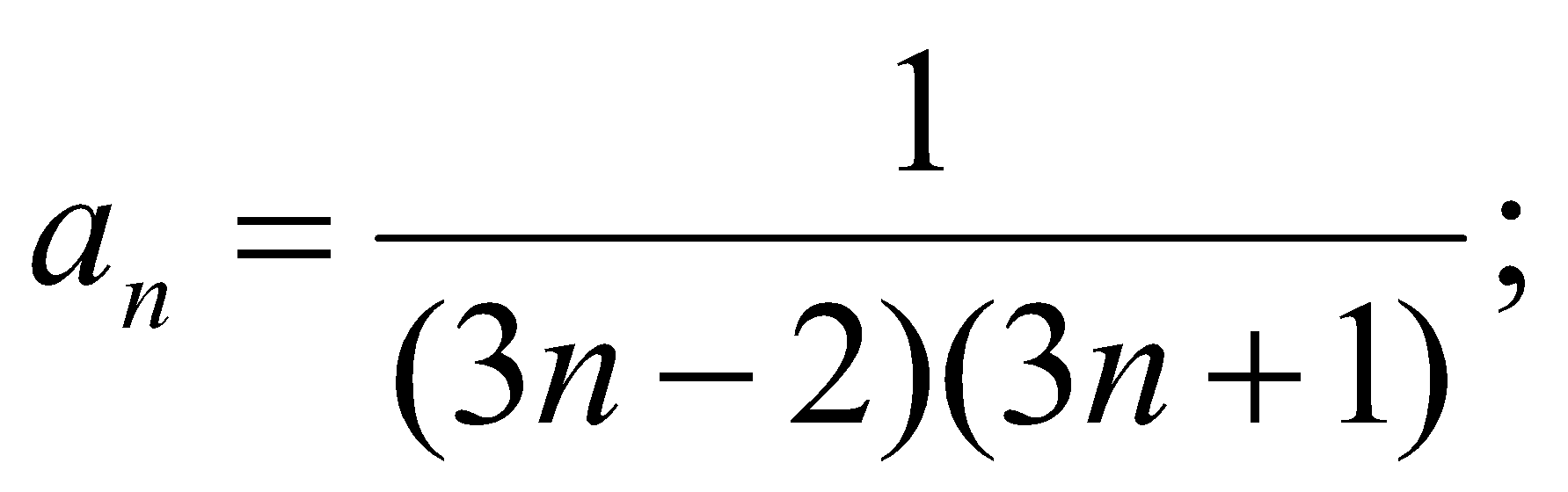
1. Дано вещественное число *B*, целое число *N* и набор из *N* вещественных чисел, упорядоченных по возрастанию. Вывести элементы набора вместе с числом *B*, сохраняя упорядоченность выводимых чисел.

***Вариант 18.***

1. Около стены наклонно стоит палка длиной х м. Один ее конец находится на расстоянии *y* м от стены. Определить значение угла α между палкой и полом для значений *x*=*k* м и *y,* изменяющегося от 2 до 3 м с шагом *h* м.
2. Дано натуральное число N. Вычислить:



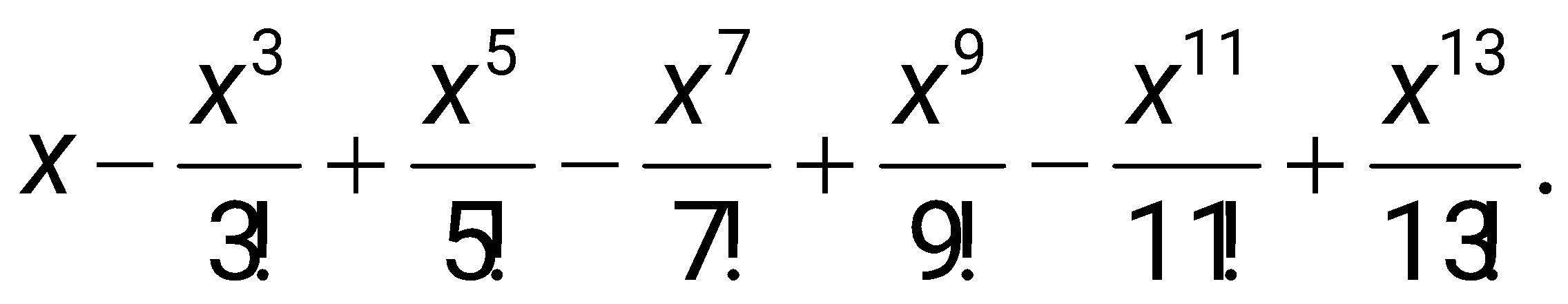
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



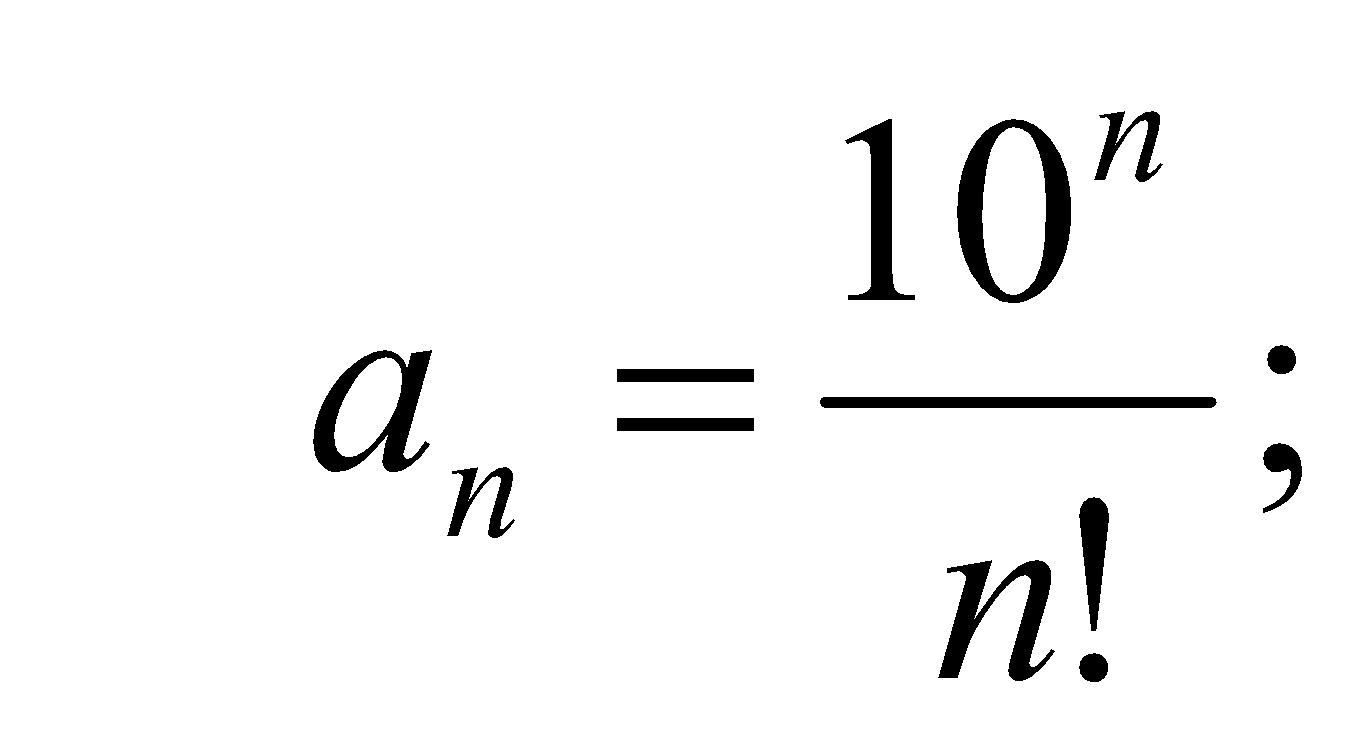
1. Дано целое число *N* и набор из *N* целых чисел, упорядоченный по возрастанию. Данный набор может содержать одинаковые элементы. Вывести в том же порядке все различные элементы данного набора.

***Вариант 19.***

1. У гусей и кроликов вместе 64 лапы. Сколько могло быть кроликов и гусей (указать все сочетания, которые возможны)?
2. Дано действительное число. Вычислить:



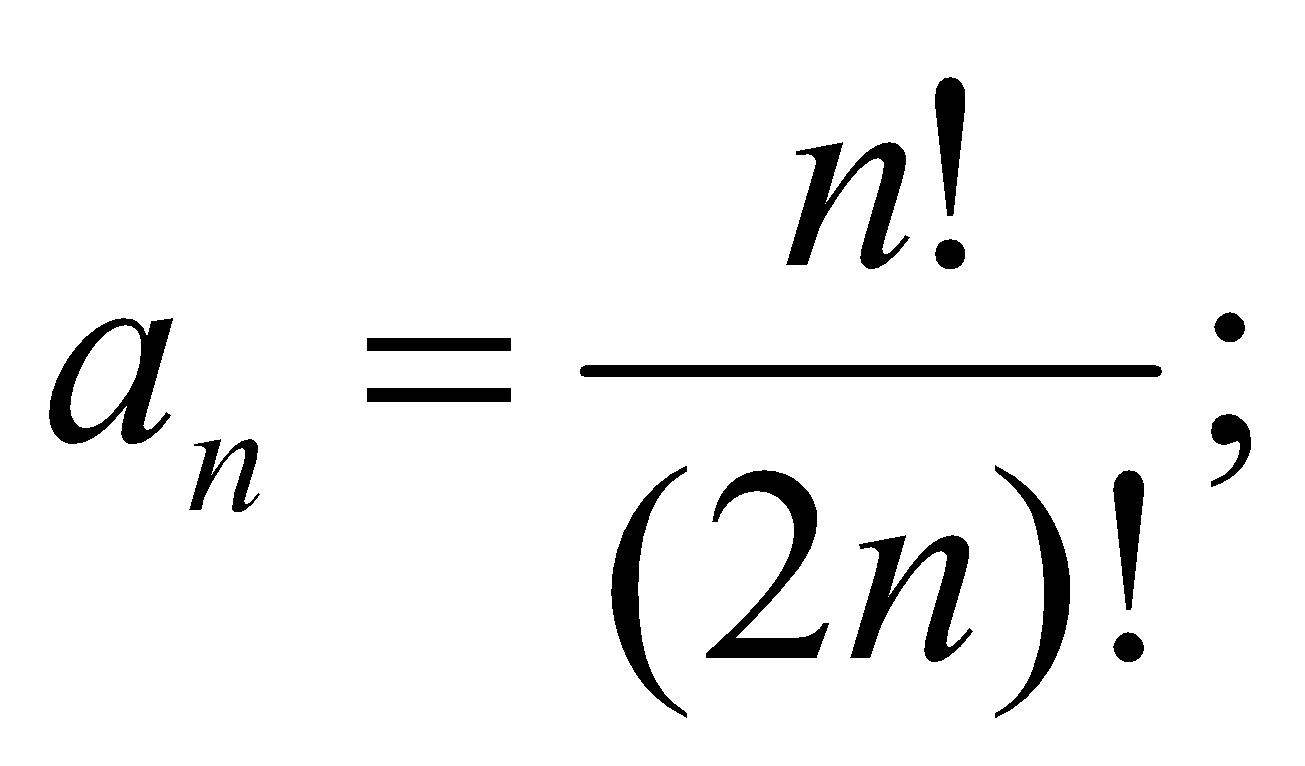
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



1. Дано целое число *N* (> 1) и набор из *N* целых чисел. Вывести те элементы в наборе, которые меньше своего левого соседа, и количество *K* таких элементов.

***Вариант 20.***

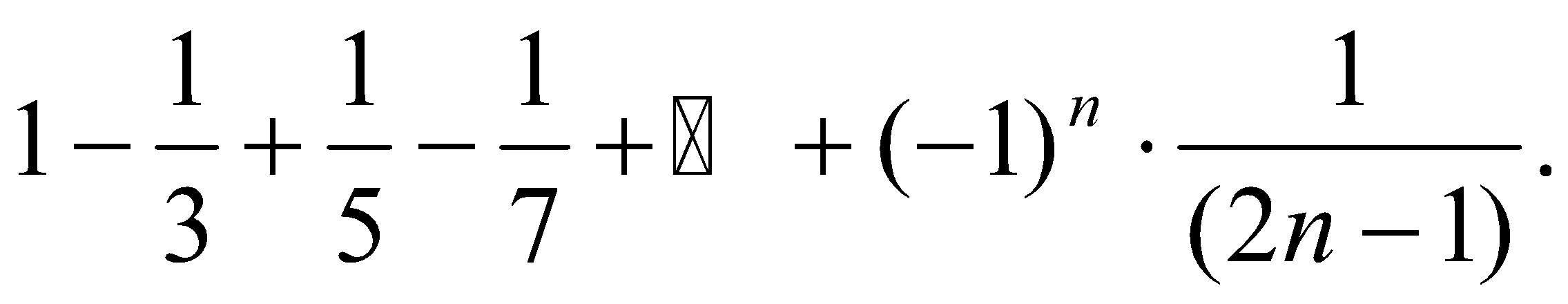
1. Составить алгоритм решения задачи: сколько можно купить быков, коров и телят, платя за быка 10 р., за корову - 5 р., а за теленка - 0,5 р., если на 100 р. надо купить 100 голов скота?
2. Даны: натуральное *n*, действительное *х*. Вычислить:
3. Найти сумму ряда с точностью ε=0,001, общий член которого:



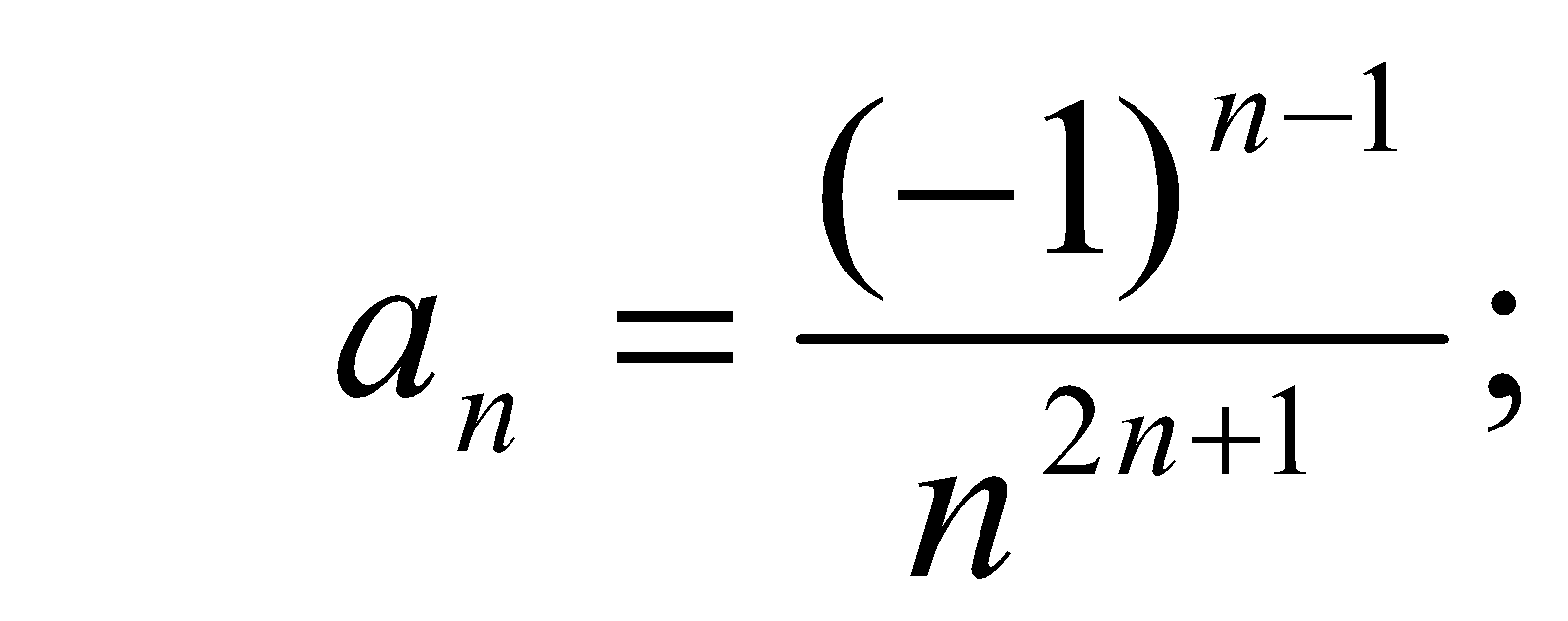
1. Дано целое число *N* (> 1) и набор из *N* целых чисел. Вывести те элементы в наборе, которые меньше своего правого соседа, и количество *K* таких элементов.

***Вариант 21.***

1. Дано натуральное число. Определить:  
   а) количество цифр 3 в нем;  
   б) сколько раз в нем встречается последняя цифра;  
   в) количество четных цифр в нем. Составное условие и более одного неполного условного оператора не использовать;
2. Дано натуральное число *N*. Вычислить:

S=

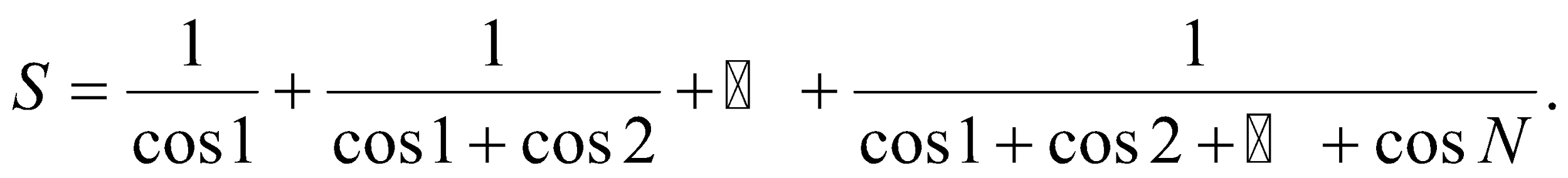
1. Найти сумму ряда с точностью ε=0,001, общий член которого:

******

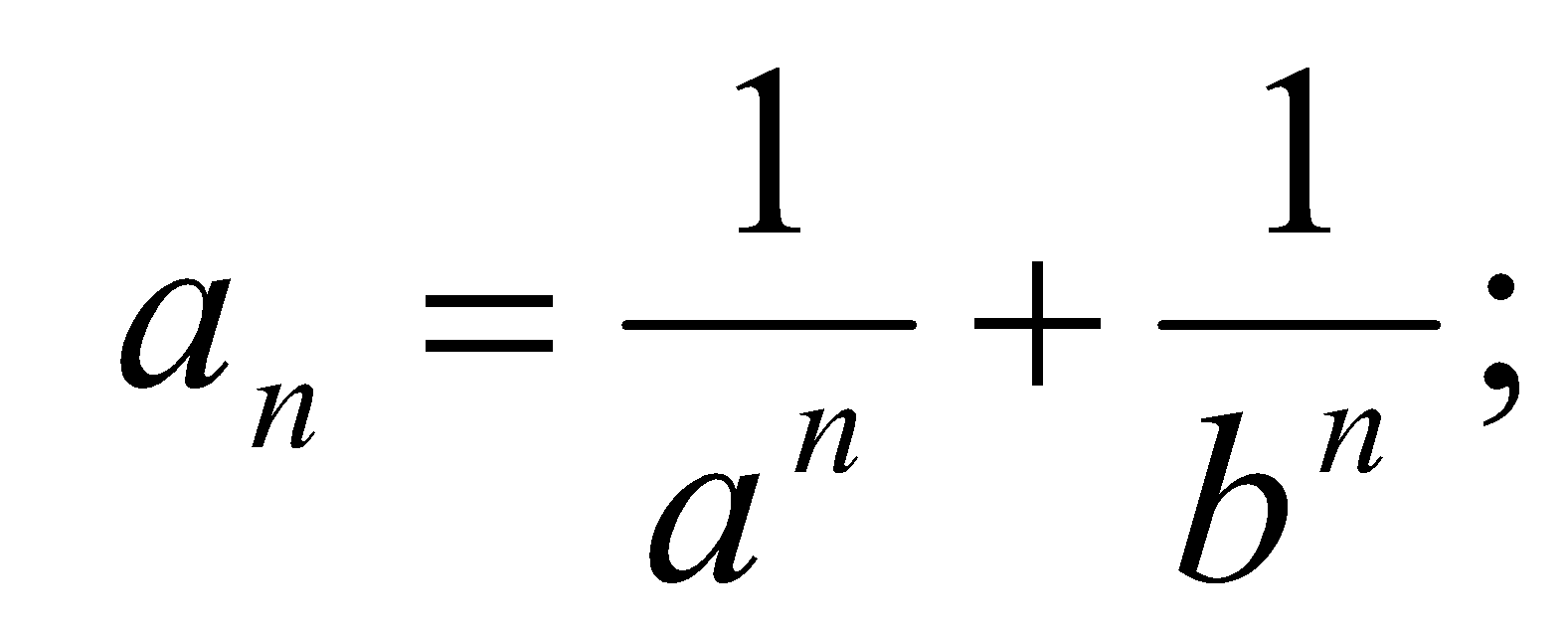
1. Дано целое число *N* (> 1) и набор из *N* вещественных чисел. Проверить, образует ли данный набор возрастающую последовательность. Если образует, то вывести True, если нет — вывести False.

***Вариант 22.***

1. Начав тренировки, спортсмен в первый день пробежал 15 км. Каждый день он увеличивал дневную норму на 13% нормы предыдущего дня. Какой суммарный путь пробежит спортсмен за 18 дней?
2. Дано натуральное число N. Вычислить:



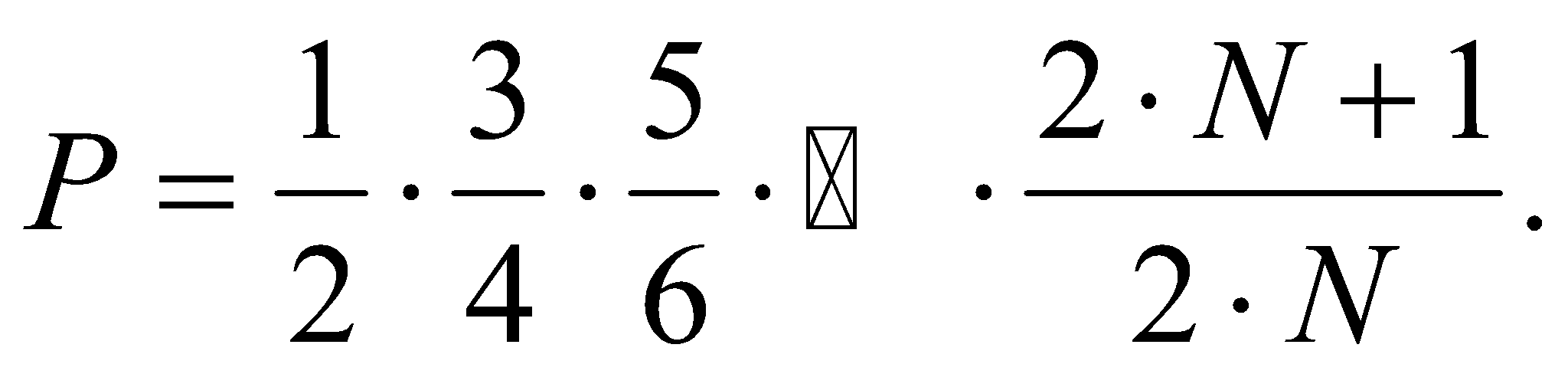
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



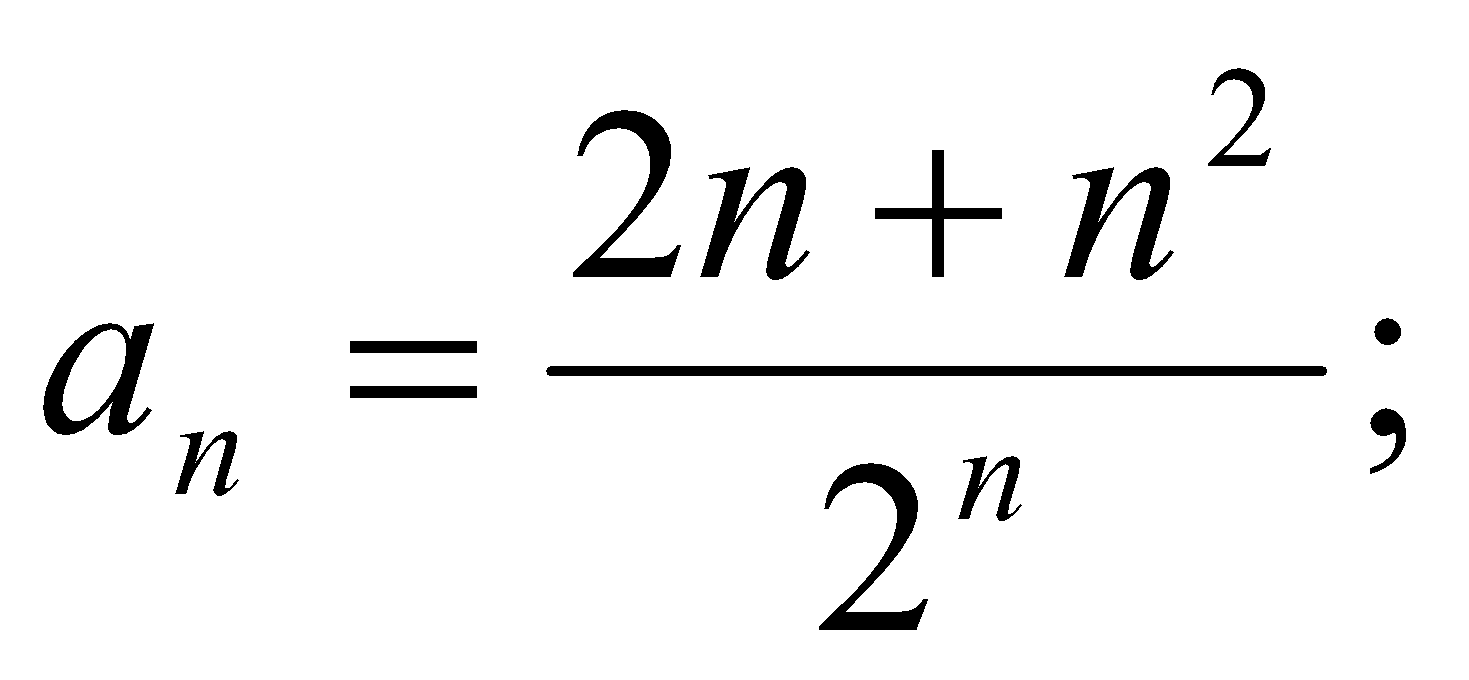
1. Дано целое число *N* (> 1) и набор из *N* вещественных чисел. Если данный набор образует убывающую последовательность, то вывести 0;   
   в противном случае вывести номер первого числа, нарушающего закономерность.

***Вариант 23.***

1. Дано натуральное число. Определить:  
   а) сколько раз в нем встречается цифра *а*;  
   б) количество его цифр, кратных *z* (значение *z* вводится с клавиатуры; *z* 2, 3, 4);  
   в) сумму его цифр, больших *a* (значение *a* вводится с клавиатуры; 0<=*a<=*8);
2. Дано натуральное число *N*. Вычислить произведение первых *N* сомножителей:



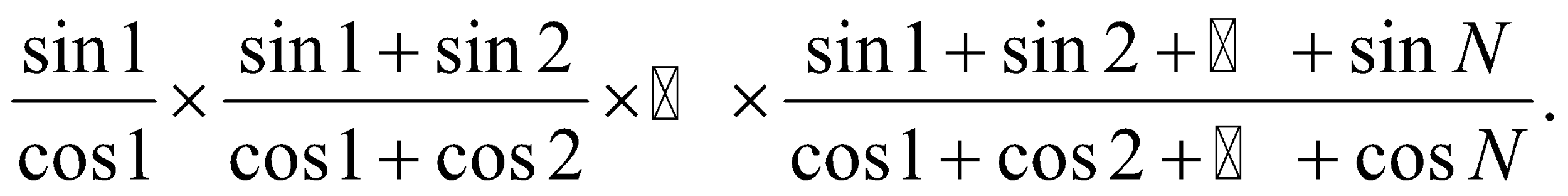
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



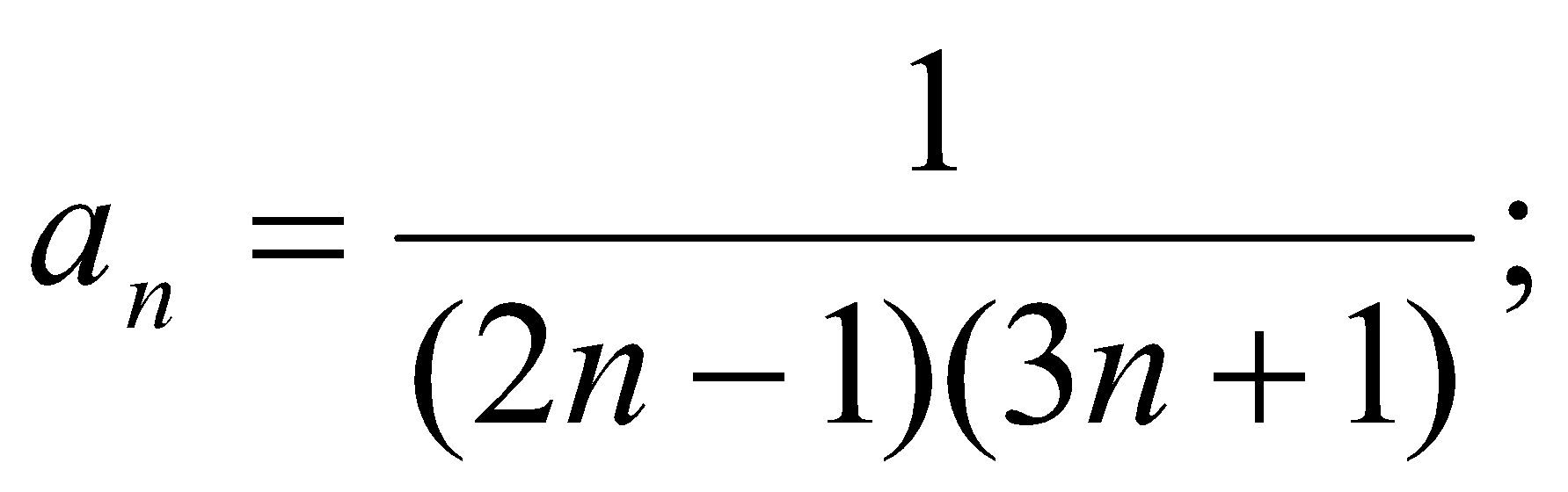
1. Дано целое число *N* (> 2) и набор из *N* вещественных чисел. Набор называется *пилообразным*, если каждый его внутренний элемент либо больше, либо меньше обоих своих соседей (то есть является «зубцом»). Если данный набор является пилообразным, то вывести 0; в противном случае вывести номер первого элемента, не являющегося зубцом.

***Вариант 24.***

1. Дана непустая последовательность целых чисел, оканчивающаяся нулем. Найти:  
   а) сумму всех чисел последовательности, больших числа *x*;  
   б) количество всех четных чисел последовательности.
2. Дано натуральное число N. Вычислить:



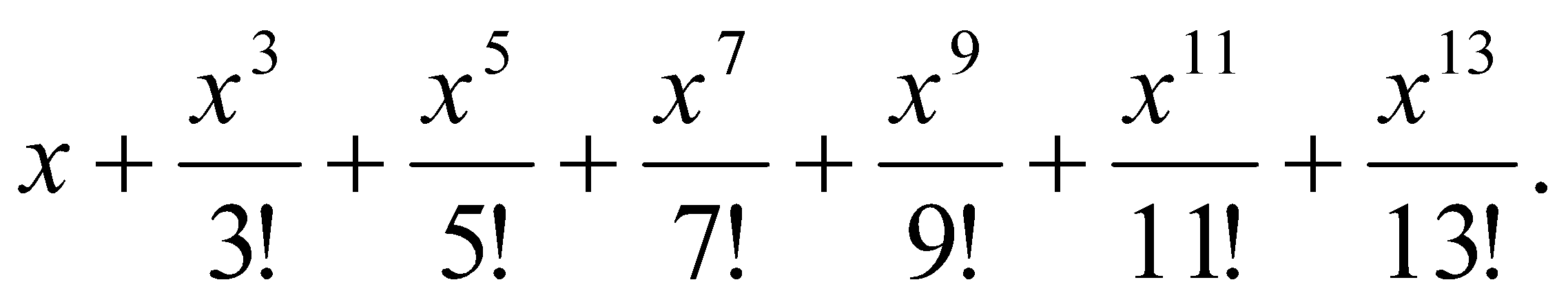
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



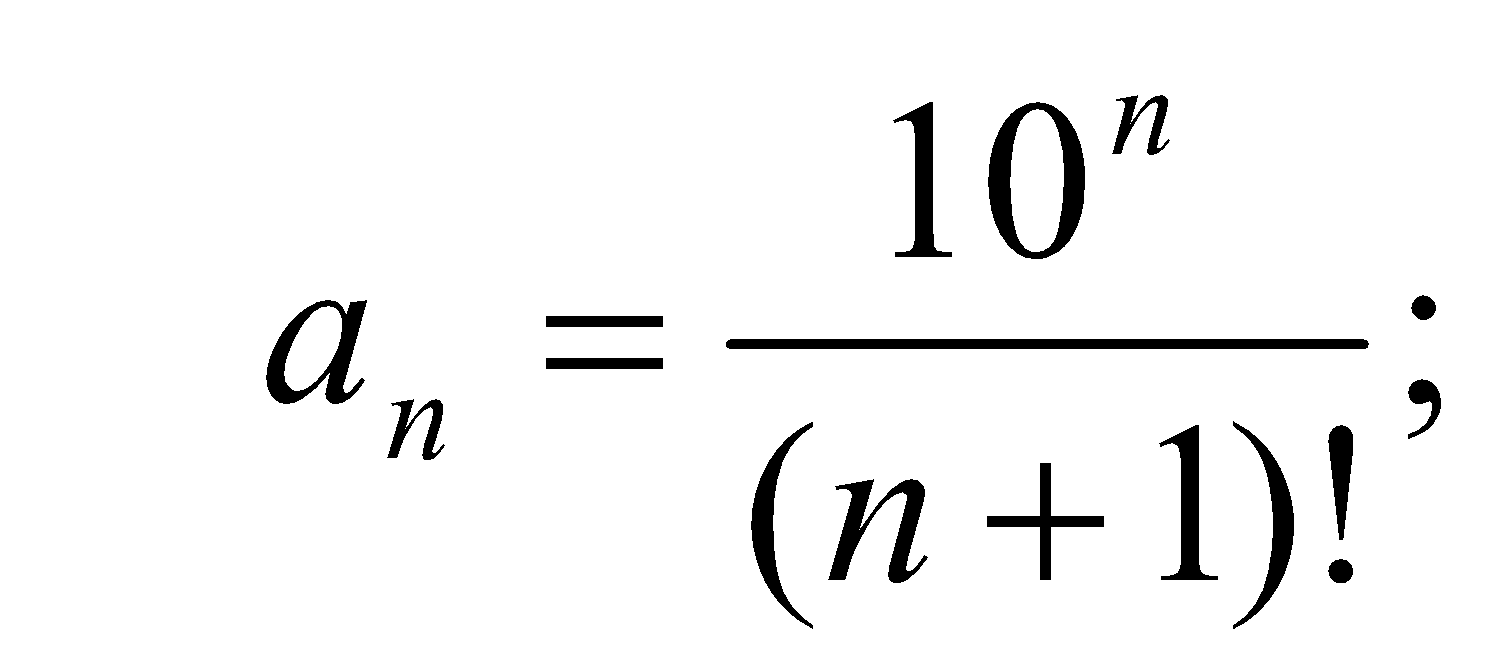
1. Дано целое число *N* и набор из *N* целых чисел, содержащий по крайней мере два нуля. Вывести сумму чисел из данного набора, расположенных между последними двумя нулями (если последние нули идут подряд, то вывести 0).

***Вариант 25.***

1. Дана последовательность ненулевых целых чисел, оканчивающая нулем. Определить, сколько раз в этой последовательности меняется знак. (Например, в последовательности 10, –4, 12, 56, –4 знак меняется 3 раза.)
2. Дано действительное число. Вычислить:



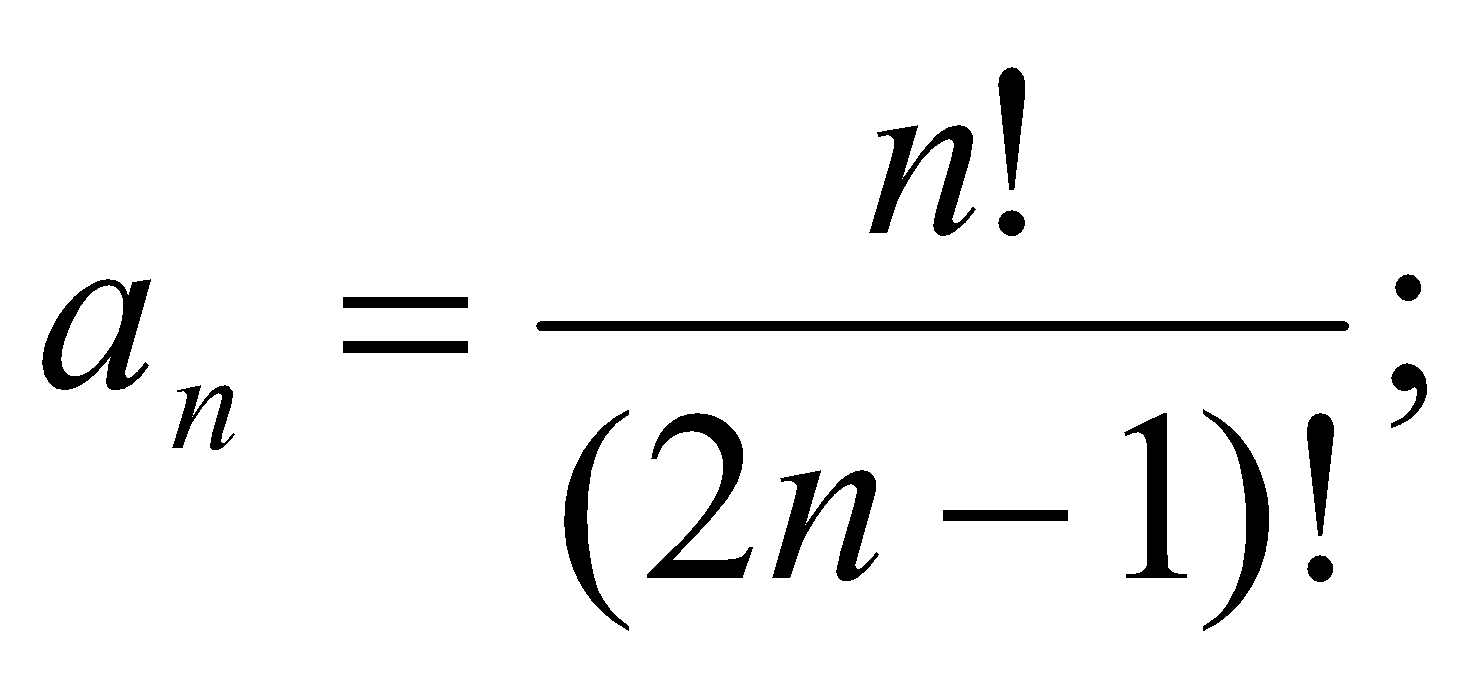
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



1. Дано целое число *N* и набор из *N* целых чисел, содержащий по крайней мере два нуля. Вывести сумму чисел из данного набора, расположенных между первым и последним нулем (если первый и последний нули идут подряд, то вывести 0).

***Вариант 26.***

1. Дано натуральное число.  
   а) Определить его максимальную цифру.  
   б) Определить его минимальную цифру.
2. Даны: натуральное *n*, действительное *х*. Вычислить:
3. Найти сумму ряда с точностью ε=0,001, общий член которого:

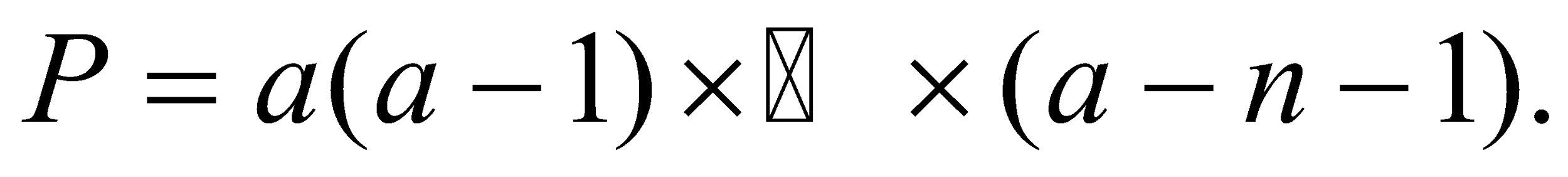


1. Даны целые числа *K*, *N* и набор из *N* вещественных чисел: *A*1, *A*2, …, *AN*. Вывести *K*-e степени чисел из данного набора:

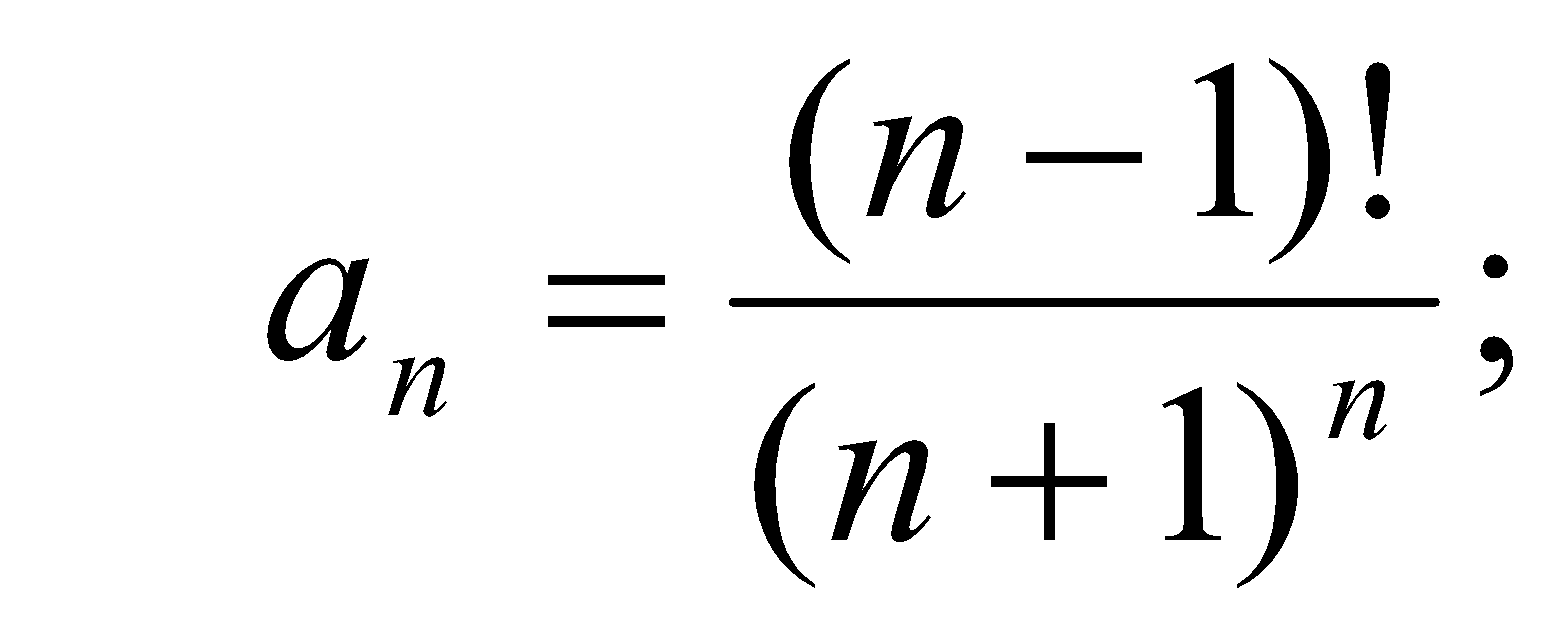
(*A*1)*K*, (*A*2)*K*, …, (*AN*)*K*.

***Вариант 27.***

1. Дано натуральное число.  
   а) Определить его максимальную и минимальную цифры.  
   б) Определить, на сколько его максимальная цифра превышает минимальную.  
   в) Найти сумму его максимальной и минимальной цифр.
2. Даны: действительное число *a*, натуральное число *n*. Вычислить:



1. Найти сумму ряда с точностью ε=10-3, общий член которого:



1. Дано целое число *N* и набор из *N* вещественных чисел: *A*1, *A*2, …, *AN*. Вывести следующие числа:

*A*1, (*A*2)2, …, (*AN*–1)*N*–1, (*AN*)*N*.

***Вариант 28.***

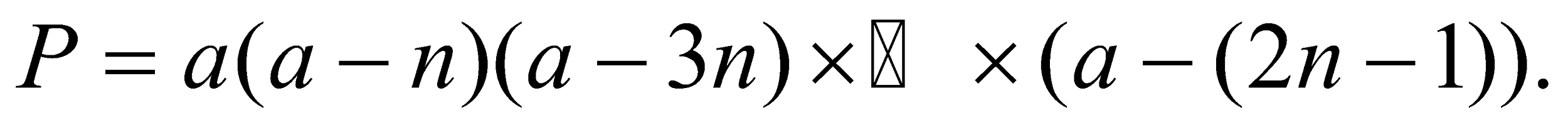
1. Дано натуральное число, в котором все цифры различны. Определить:  
   а) порядковый номер его максимальной цифры, считая номера:

* от конца числа;
* от начала числа;

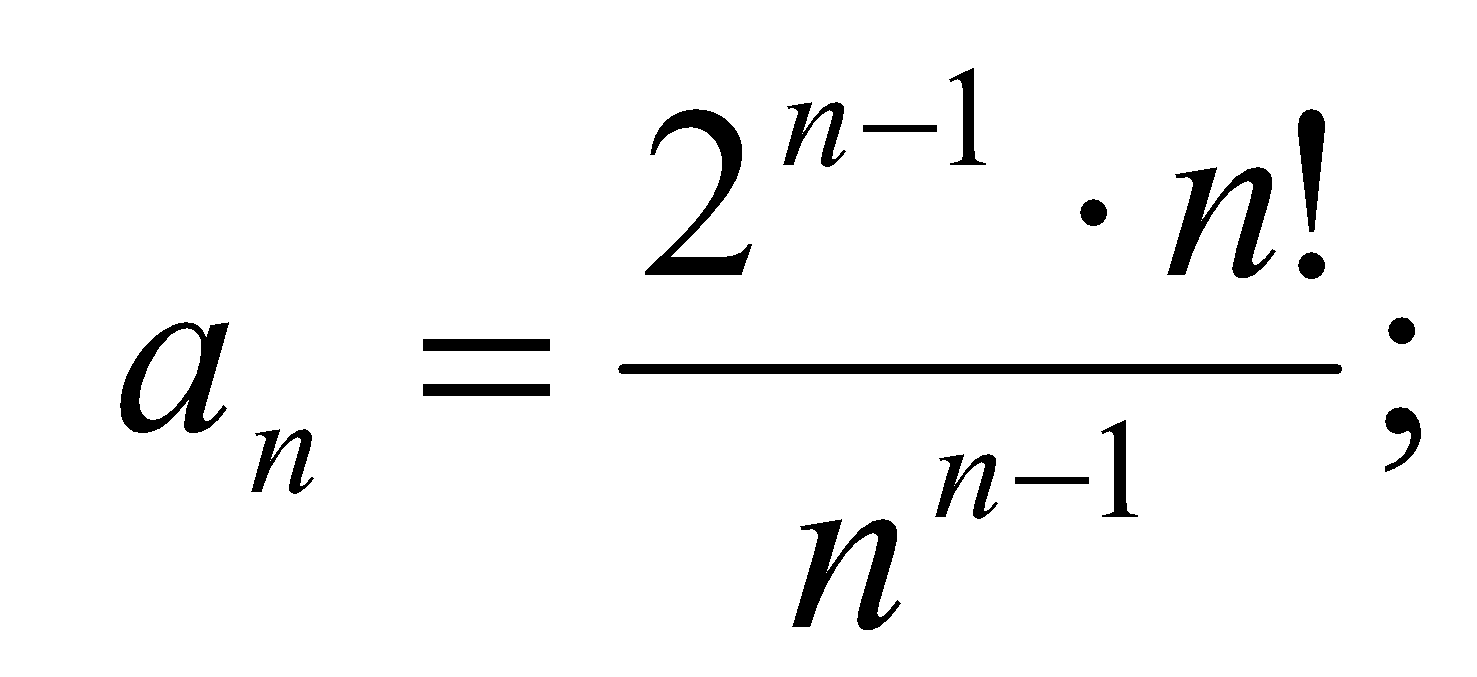
б) порядковый номер его минимальной цифры, считая номера:

* от конца числа;
* от начала числа.

1. Даны: действительное число *a*, натуральное число *n*. Вычислить:



1. Найти сумму ряда с точностью ε=0,001, общий член которого:

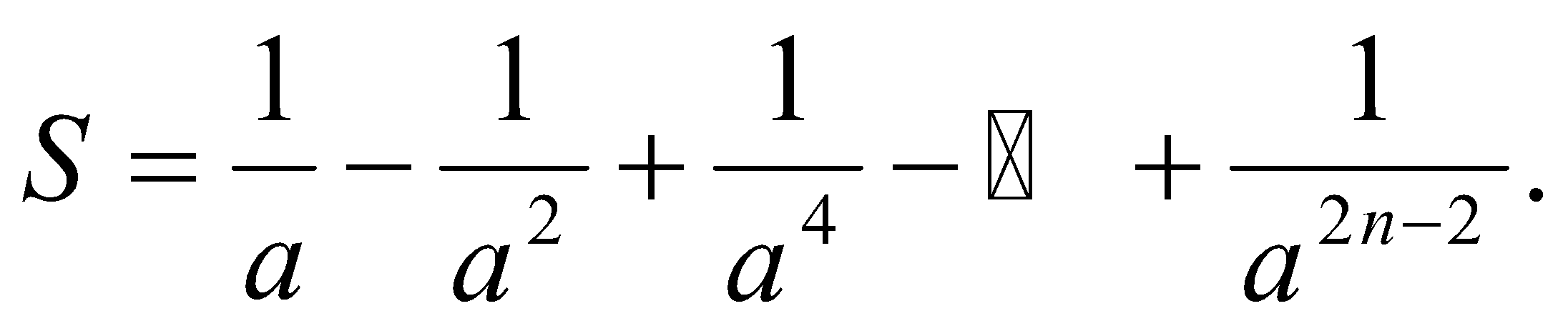


1. Дано целое число *N* и набор из *N* вещественных чисел: *A*1, *A*2, …, *AN*. Вывести следующие числа:

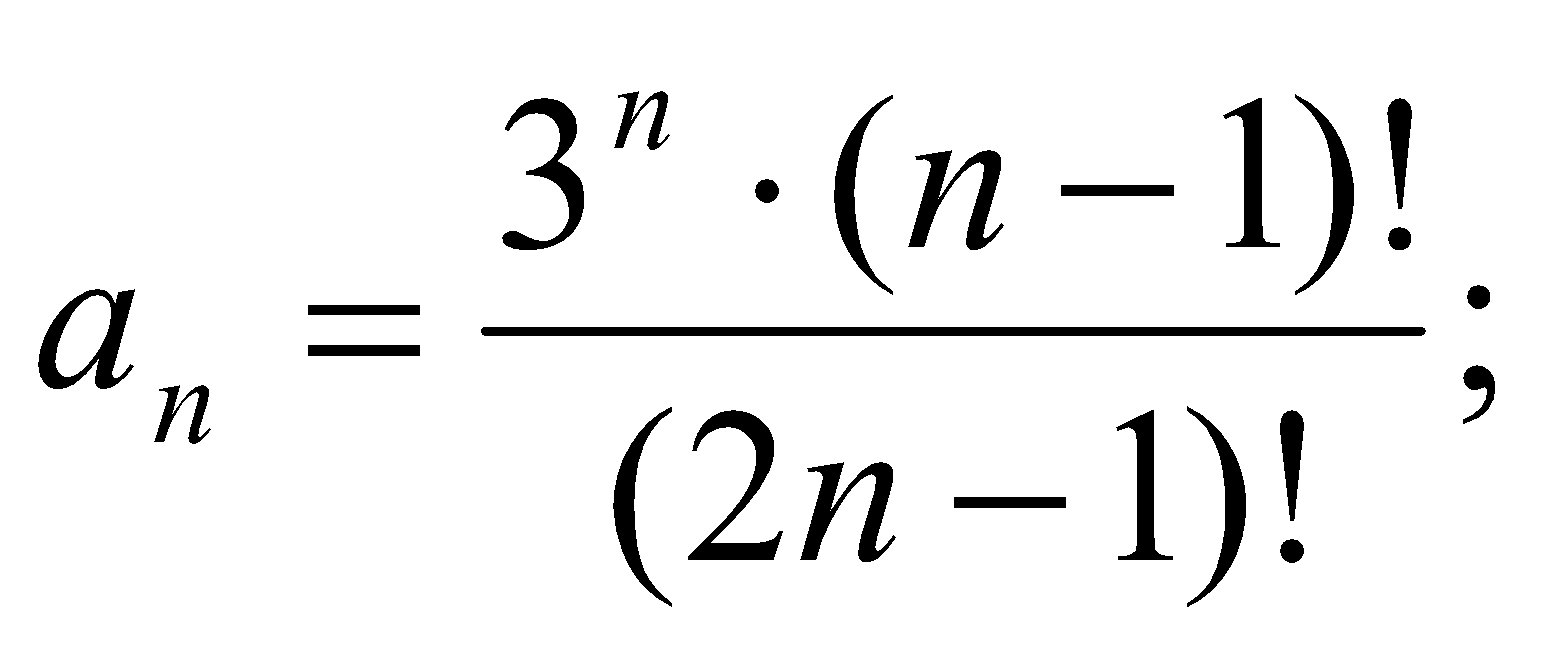
(*A*1)*N*, (*A*2)*N*–1, …, (*AN*–1)2, *AN*.

***Вариант 29.***

1. Дано натуральное число. Определить номер цифры 8 в нем, считая от конца  
   числа. Если такой цифры нет, ответом должно быть число 0, если таких цифр  
   в числе несколько *—* должен быть определен номер самой левой из них.
2. Даны: действительное число *а*, натуральное число *п*. Вычислить:



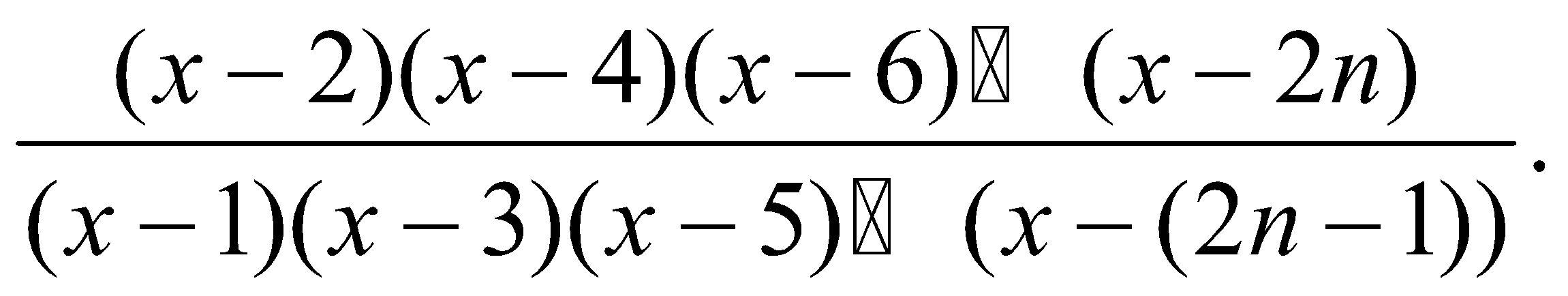
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



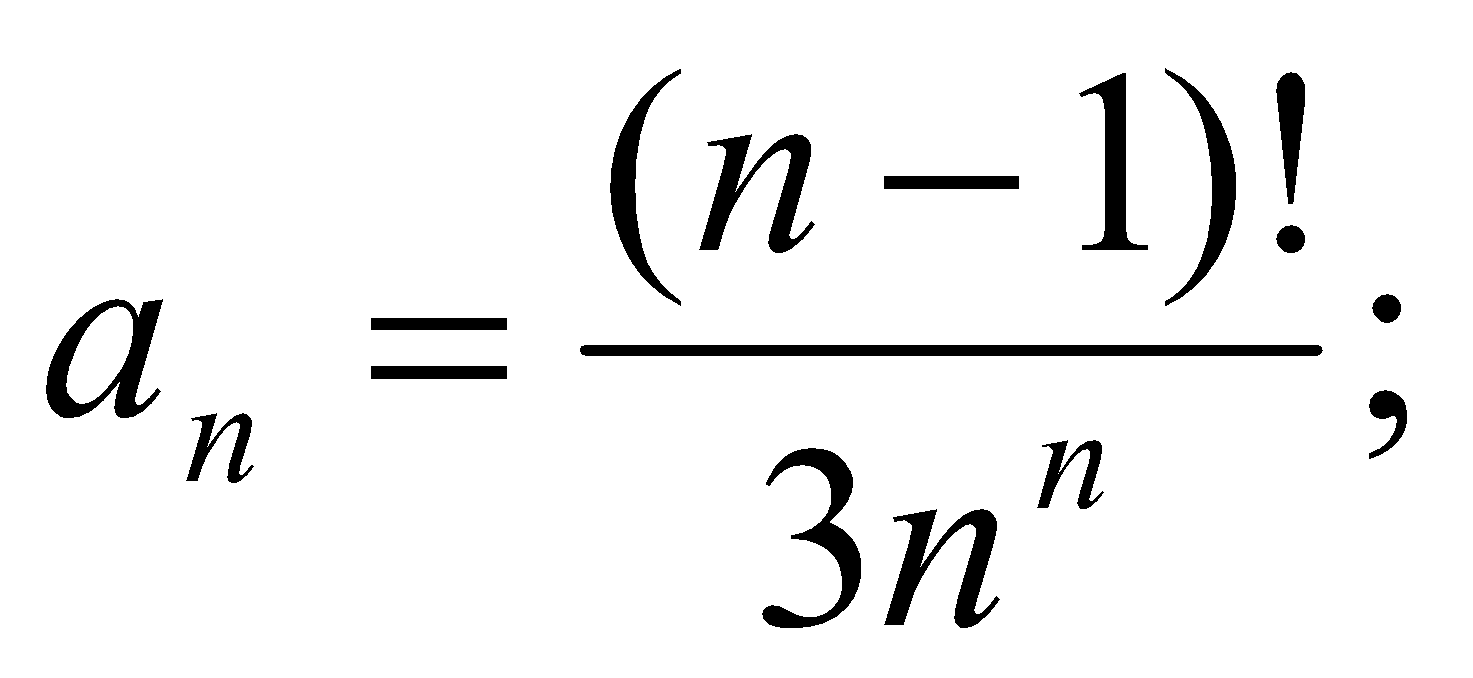
1. Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Вывести общую сумму всех элементов, входящих в данные наборы.

***Вариант 30.***

1. Дано натуральное число. Определить, сколько раз в нем встречается максимальная цифра (например, для числа 132 233 ответ равен 3, для числа  
   46 336 *—* 2, для числа 12 345 *—* 1).
2. Дано действительное *x*. Вычислить:



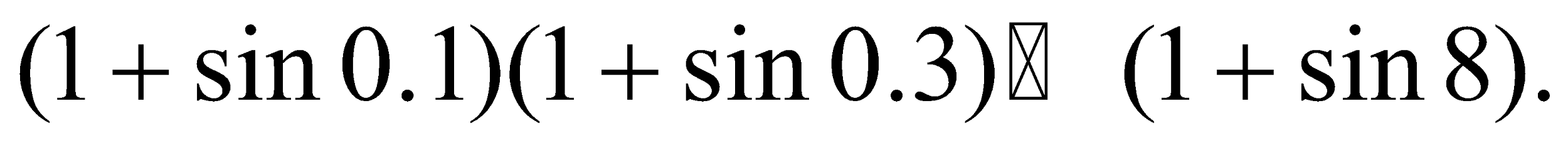
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



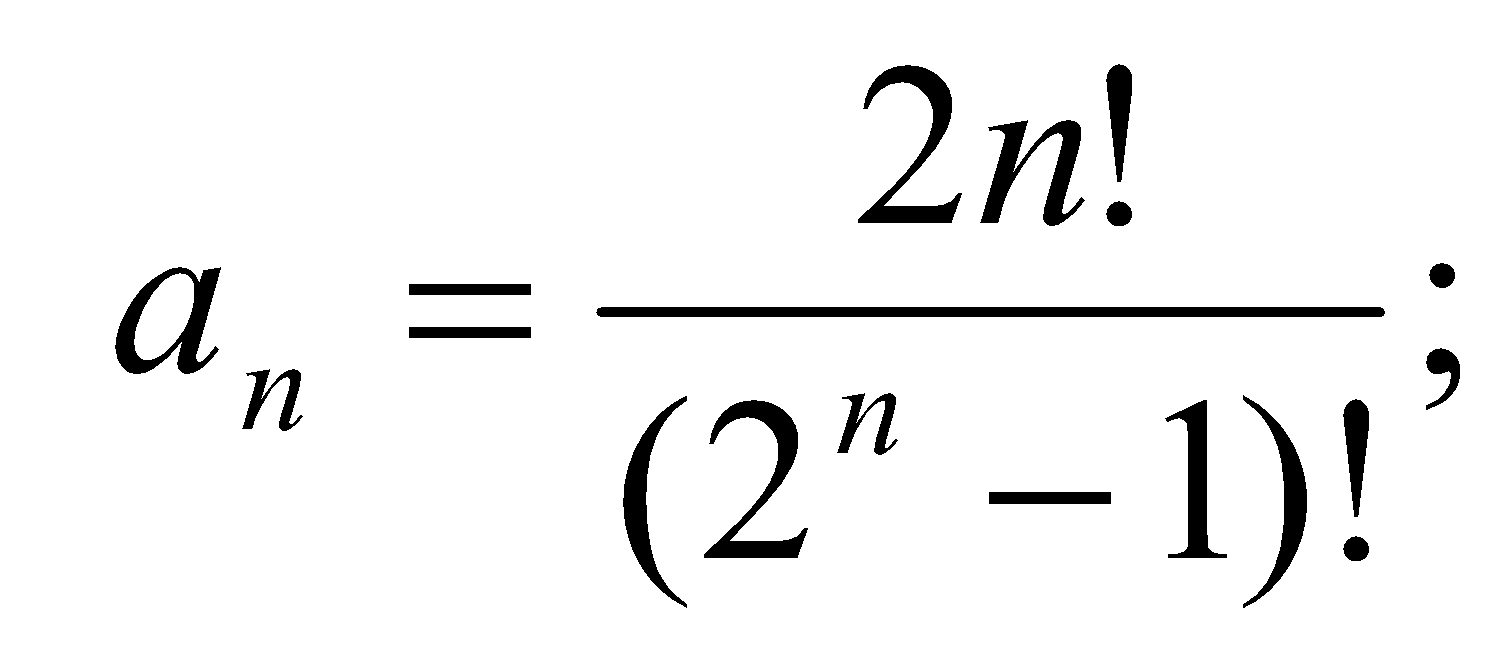
1. Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Для каждого набора вывести сумму его элементов.

***Вариант 31.***

1. Дано натуральное число. Определить, сколько раз в нем встречается минимальная цифра (например, для числа для числа 102 200 ответ равен 3, для  
   числа 40 330 *—* 2, для числа 10 345 *—* 1).
2. Вычислить:



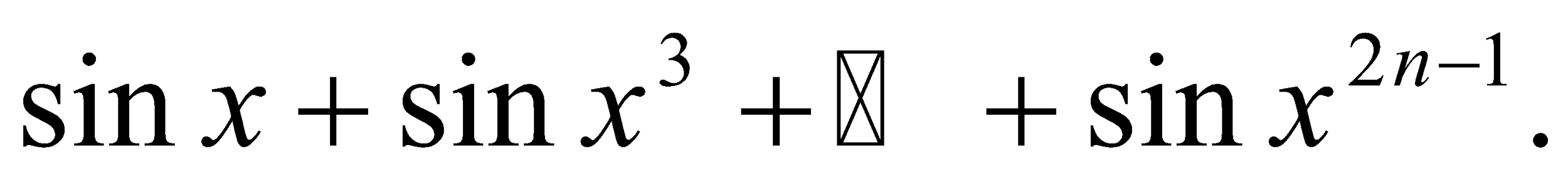
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



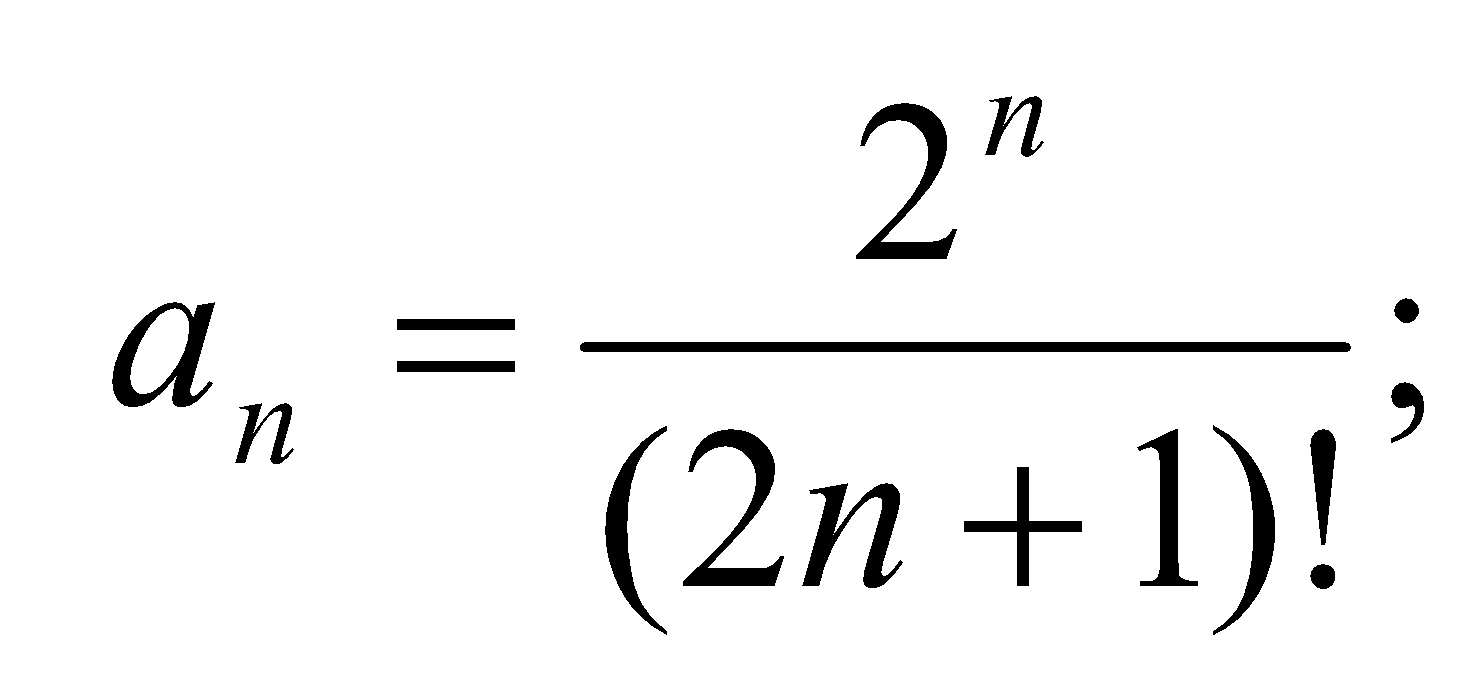
1. Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Найти количество наборов, содержащих число 2. Если таких наборов нет, то вывести 0.

***Вариант 32.***

1. Напечатать все кратные тринадцати натуральные числа, меньшие 100. Задачу решить двумя способами:  
   а) без использования оператора цикла с условием;  
   б) с использованием оператора цикла с условием.
2. Даны натуральное *п*, действительное *х*. Вычислить:



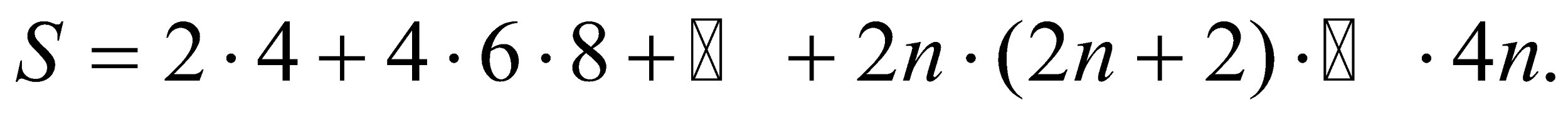
1. Найти сумму ряда с точностью ε=10-3, общий член которого:

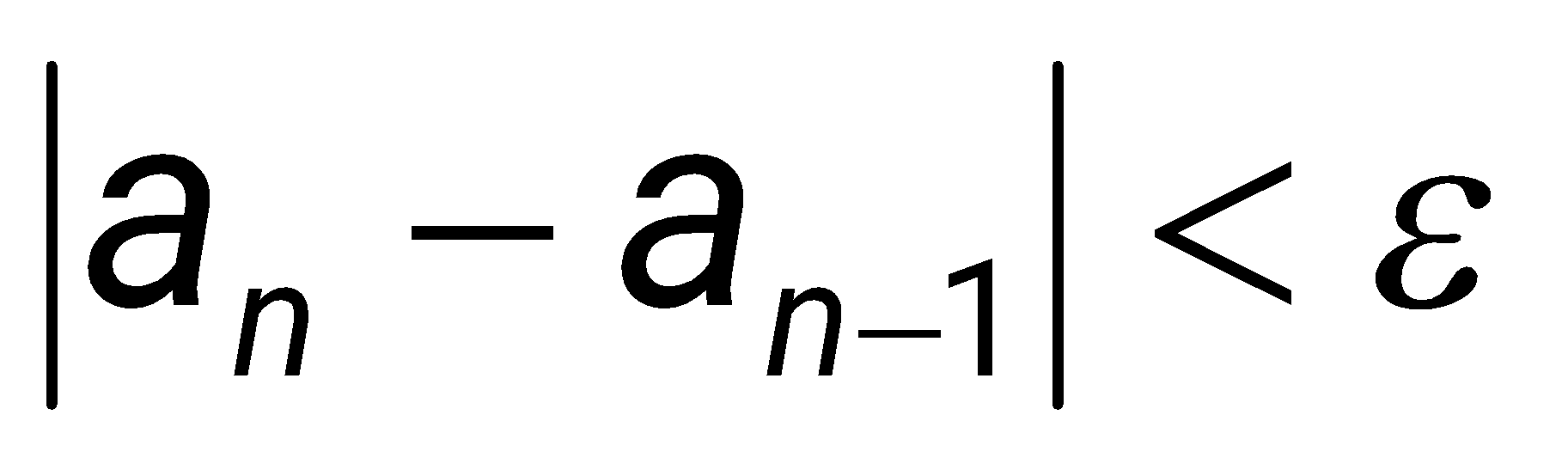
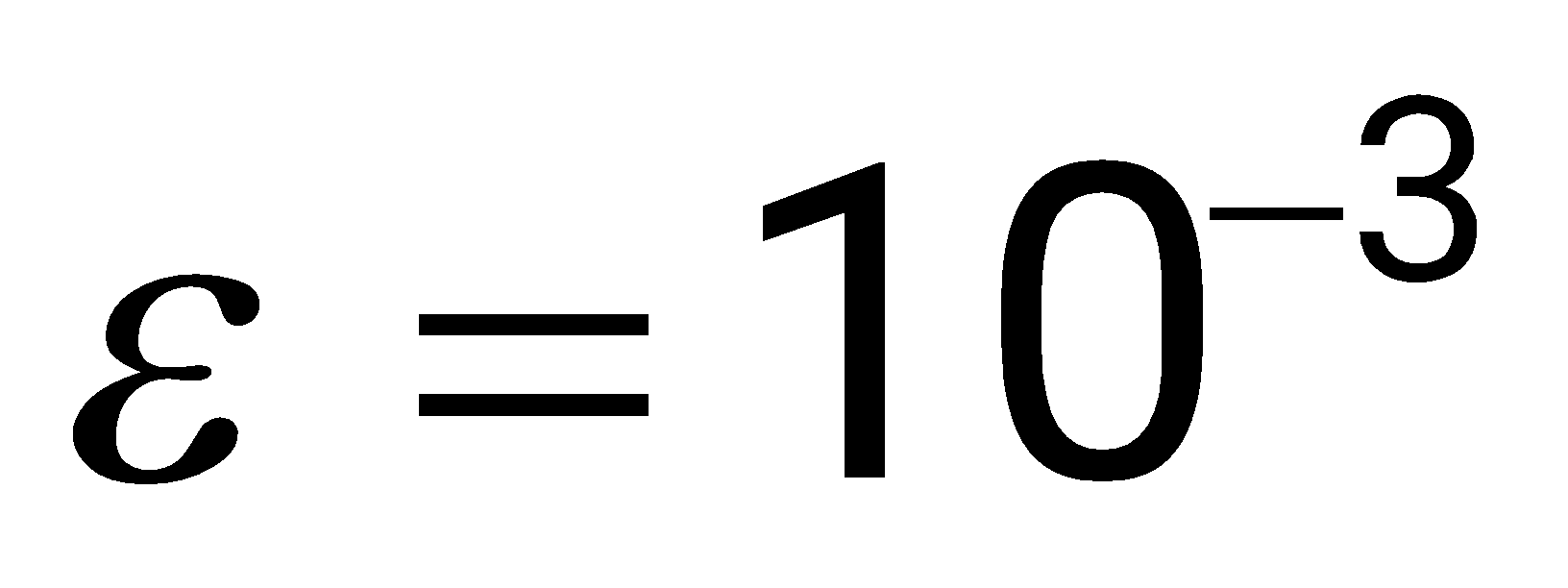
******

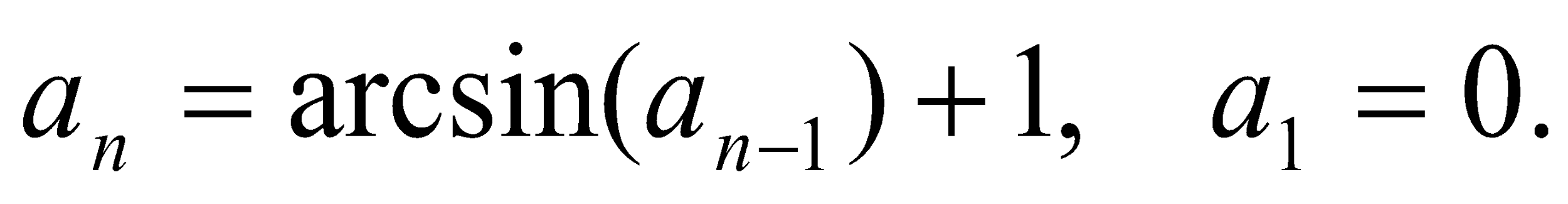
1. Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Для каждого набора вывести номер его первого элемента, равного 2, или число 0, если в данном наборе нет двоек.

***Вариант 33.***

1. Найти 15 первых натуральных чисел, делящихся нацело на 19 и находящихся  
   в интервале, левая граница которого равна 100.
2. Дано натуральное *п*. Вычислить:



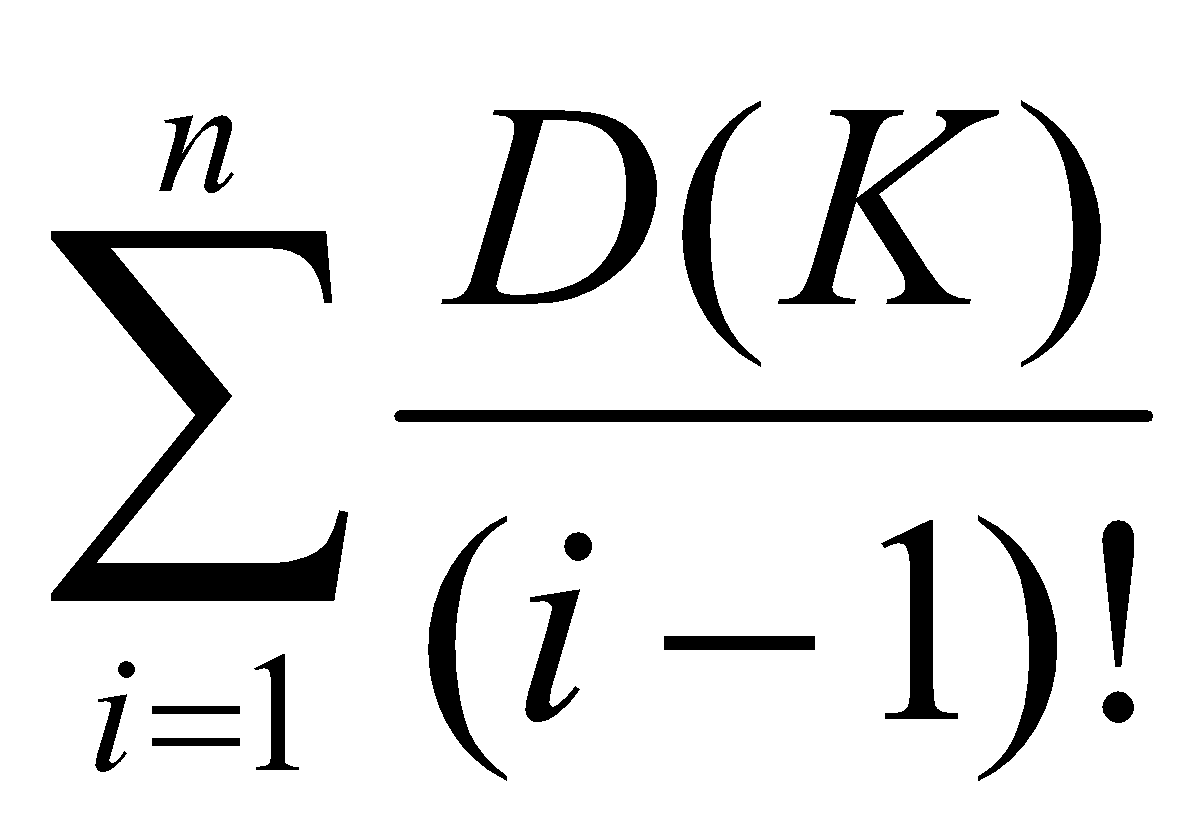
1. Найти наименьший номер последовательности, для которого выполняется условие . Вывести на экран этот номер и все элементы *ai*, где *i*=1,2,...,*п*, если 



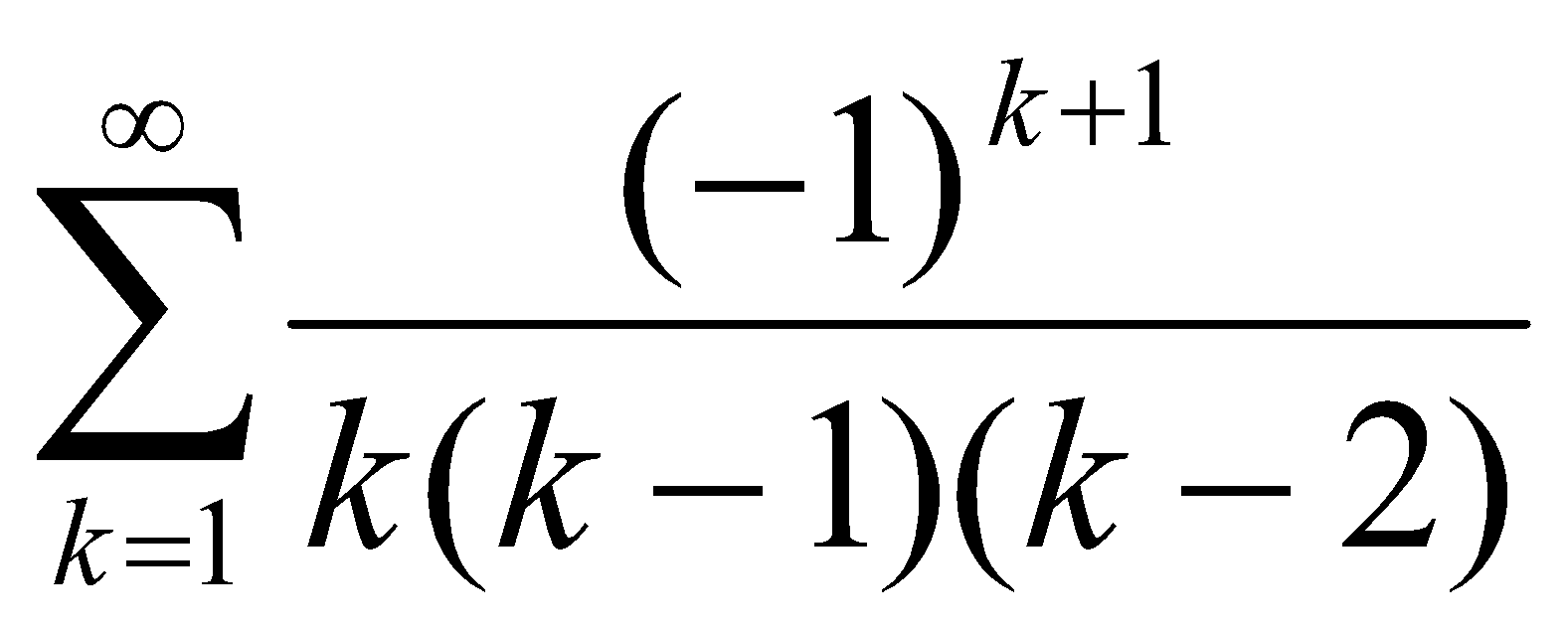
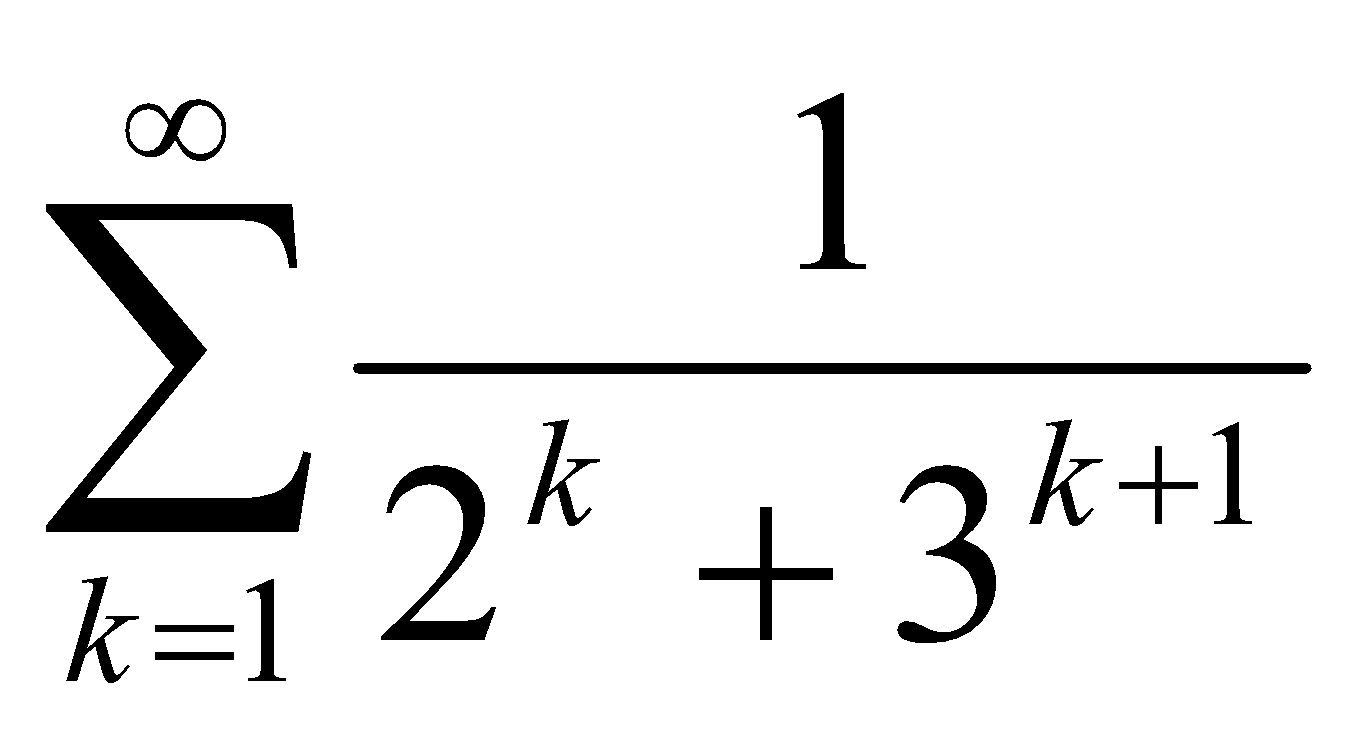
1. Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Для каждого набора вывести номер его последнего элемента, равного 2, или число 0, если в данном наборе нет двоек.

***Вариант 34.***

1. Найти 20 первых натуральных чисел, делящихся нацело на 13 или на 17 и находящихся в интервале, левая граница которого равна 500.
2. Для любого целого *К* обозначим количество цифр в его десятичной записи D(***К***). Дано натуральное число n**.** Вычислить



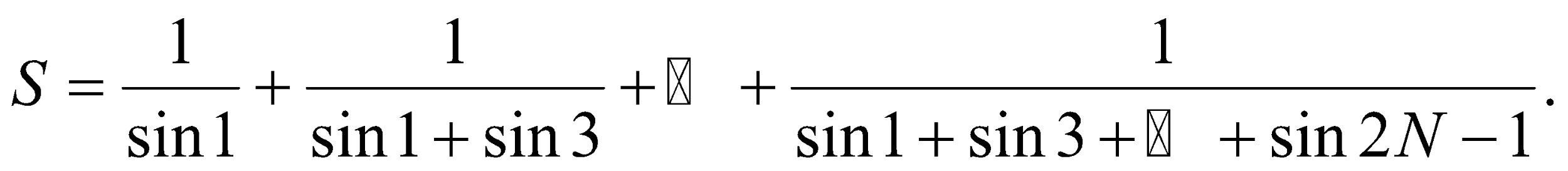
1. Вычислить бесконечную сумму с заданной точностью Eps (Eps>0). Требуемая точность достигнута, если очередное слагаемое оказалось по модулю меньше Eps. Вычислить

a)  b) 

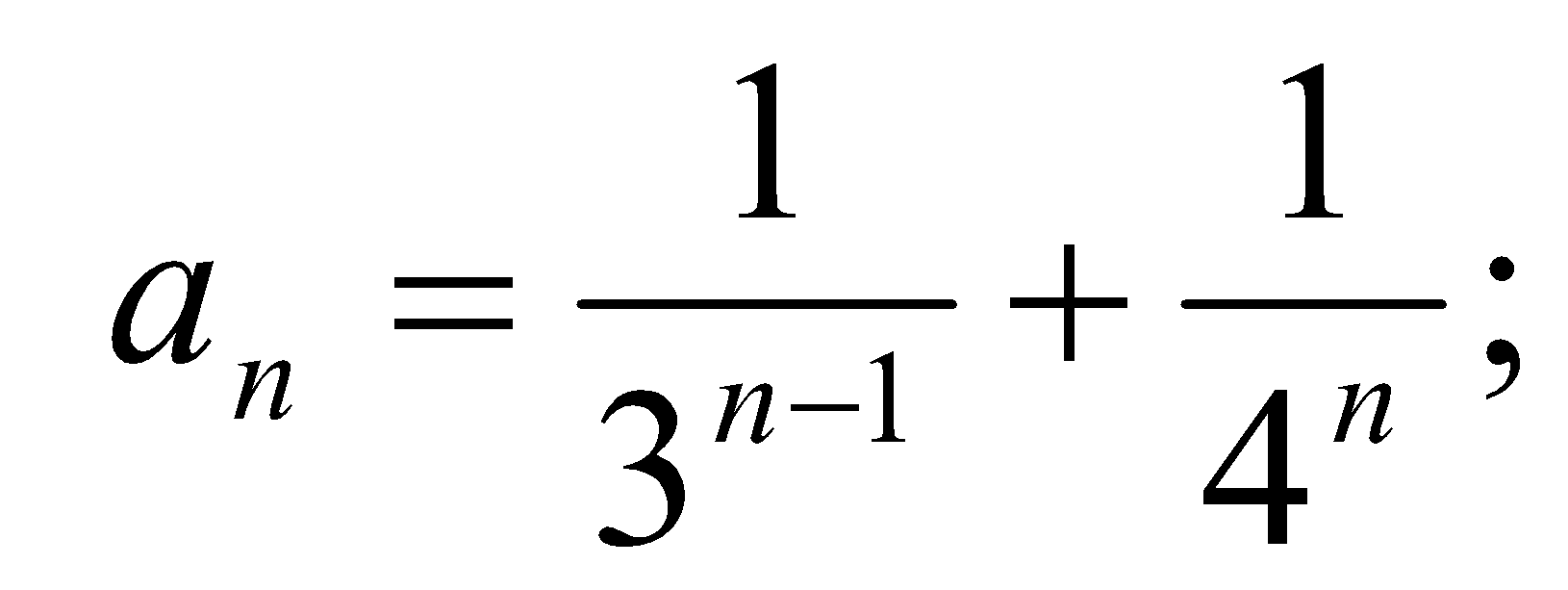
1. Даны целые числа *K*, *N*, а также *K* наборов целых чисел по *N* элементов в каждом наборе. Для каждого набора выполнить следующее действие: если в наборе содержится число 2, то вывести сумму его элементов; если в наборе нет двоек, то вывести 0.

***Вариант 35.***

1. Найти 10 первых натуральных чисел, оканчивающихся на цифру 7, кратных  
   числу 9 и находящихся в интервале, левая граница которого равна 100
2. Дано натуральное число N. Вычислить:



1. Найти сумму ряда с точностью ε=0,001, общий член которого:



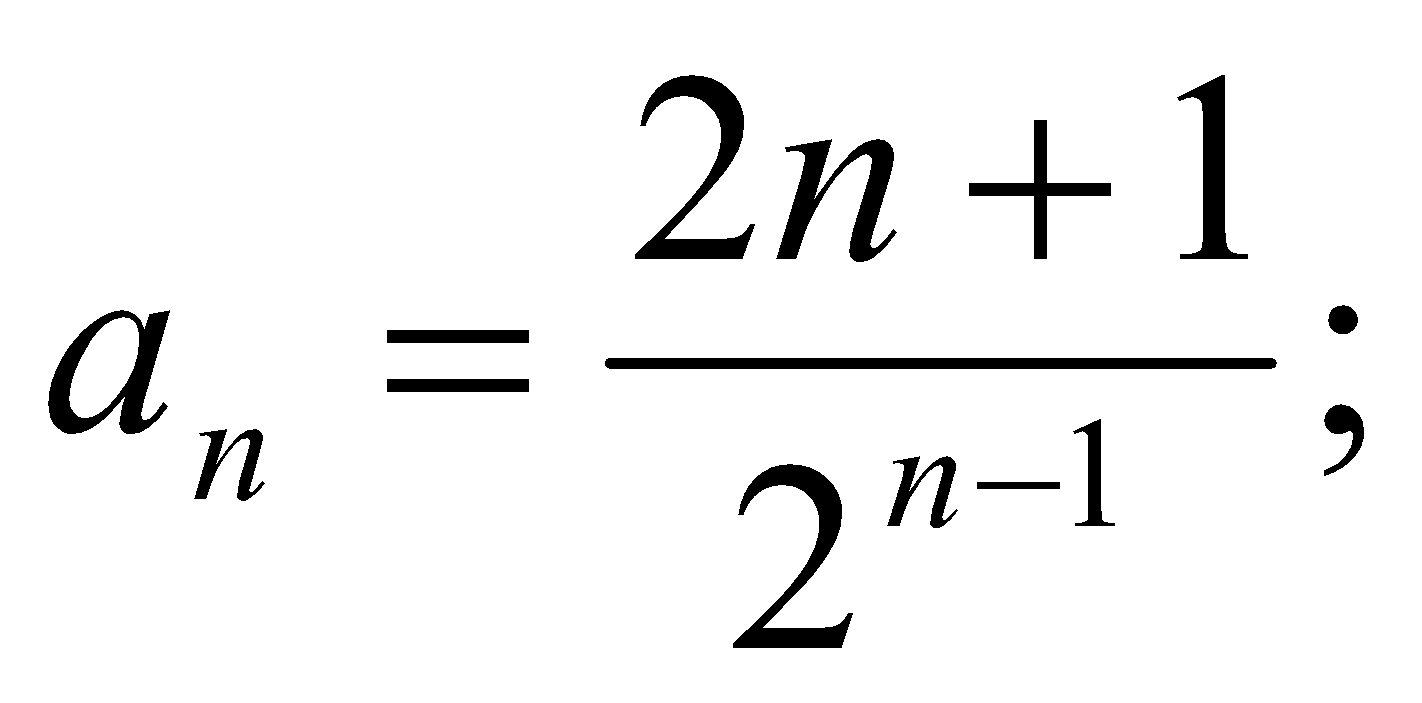
1. Дано целое число *K*, а также *K* наборов ненулевых целых чисел. Каждый набор содержит не менее двух элементов, признаком его завершения является число 0. Найти количество наборов, элементы которых возрастают.

***Вариант 36.***

1. Составить программу, определяющую общее число удалений и общее штрафное время каждой из хоккейных команд во время игры (игроки удаляются на  
   2, 5 или 10 мин). Окончание игры моделировать вводом числа 0.
2. Дано натуральное число *N*. Вычислить произведение первых *N* сомножителей:



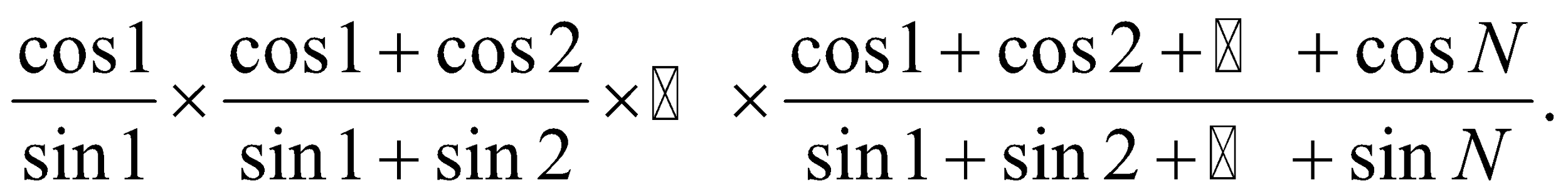
1. Найти сумму ряда с точностью ε=10-3, общий член которого:



1. Дано целое число *K*, а также *K* наборов ненулевых целых чисел. Каждый набор содержит не менее двух элементов, признаком его завершения является число 0. Найти количество наборов, элементы которых возрастают или убывают.

***Вариант 37.***

1. Дана непустая последовательность вещественных чисел, оканчивающаяся  
   числом 1000. Последовательность является неубывающей. Несколько чисел,  
   идущих подряд, равны между собой. Найти количество таких чисел. Сколько  
   различных чисел имеется в последовательности?
2. Дано натуральное число N. Вычислить:



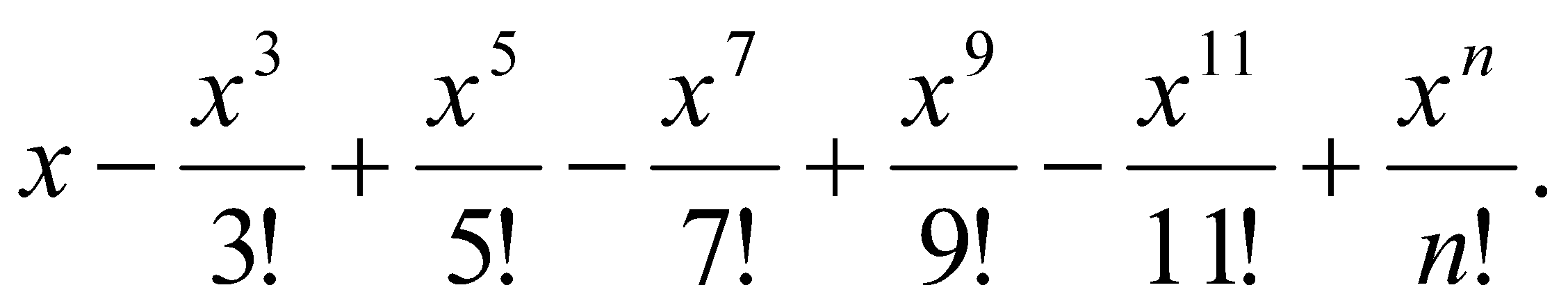
1. Найти сумму ряда с точностью ε=0,001, общий член которого:



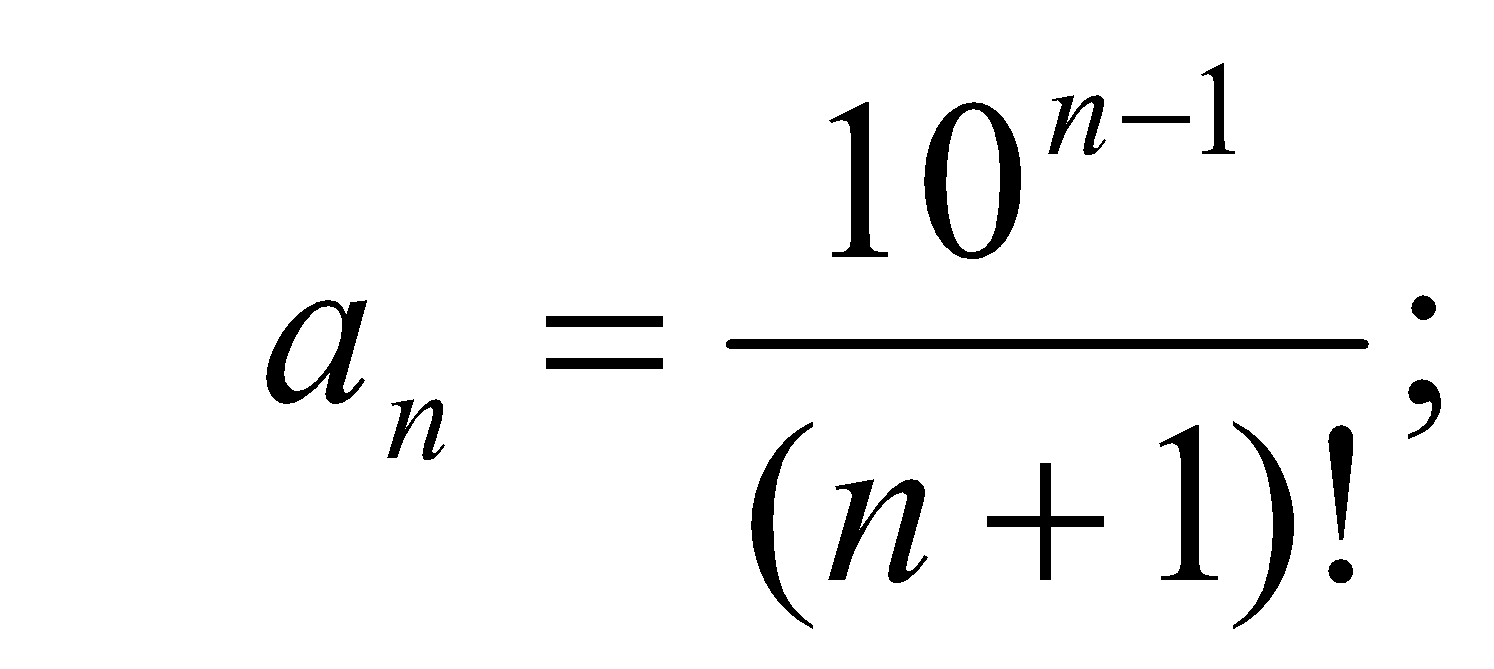
1. Дано целое число *K*, а также *K* наборов ненулевых целых чисел. Признаком завершения каждого набора является число 0. Для каждого набора вывести количество его элементов. Вывести также общее количество элементов во всех наборах.

***Вариант 38.***

1. Дана непустая последовательность вещественных чисел, оканчивающаяся нулем. Последовательность является невозрастающей. Найти количество различных чисел в последовательности
2. Дано действительное число. Вычислить:



1. Найти сумму ряда с точностью ε=10-3, общий член которого:



1. Дано целое число *K*, а также *K* наборов ненулевых целых чисел. Каждый набор содержит не менее двух элементов, признаком его завершения является число 0. Для каждого набора выполнить следующее действие: если элементы набора возрастают, то вывести 1; если элементы набора убывают, то вывести –1; если элементы набора не возрастают и не убывают, то вывести 0.

# **Процедуры и функции: группа Proc**

## **Процедуры с числовыми параметрами**

1. Описать процедуру PowerA3(*A*, *B*), вычисляющую третью степень числа *A* и возвращающую ее в переменной *B* (*A* — входной, *B* — выходной параметр; оба параметра являются вещественными). С помощью этой процедуры найти третьи степени пяти данных чисел.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **procedure** PowerA3(A: Real; **var** B: Real);  **begin**  B := A \* A \* A;  **end**;  **var**  i: Integer;  A, B: Real;  **begin**  **for** i := 1 **to** 5 **do**  **begin**  Write('A: ');  Readln(A);  PowerA3(A, B);  Writeln(B);  **end**;  **end**. | | void poweraA3(float a, float &b){  b=a\*a\*a;  }  int main(){  int i;  for(i=1;i<=5;++i){  float a,b;  cin>>a>>b  poweraA3(a, b);  cout<<b<<" ";  }  return 0;  } | |
| Python | C# | | |
| def powerA3(x):  return a\*a\*a  for i in range(0,5):  a=float(input("a"))  print(powerA3(a)) | static void powerA3(float a, out float b){  b = a \* a \* a;  }  static void Main(string[] args){  float a, b;  for (int i=0; i<5; i++){  a=Convert.ToSingle(Console.ReadLine());  powerA3(a, out b);  Console.WriteLine(b);  }  Console.ReadKey();  } | | |
| Java | | | Go |
| static void PowerA3(double a){  double b=a\*a\*a;  System.out.println(b);  }  public static void main(String[] args){  Scanner sc=new Scanner(System.in);  double a, S = 0;  for (int i=0; i<5; i++){  a=sc.nextDouble();  PowerA3(a);  }  } | | | package main  import "fmt"  func PowerA3(a float32) {  var b=a\*a\*a;  fmt.Println(b)  }  func main() {  var a float32  for i:=0; i<5; i++ {  fmt.Scan(&a)  PowerA3(a)  }  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** powerA3(a: Double) {  **var** b=Math.pow(a,3.0)  *println*(b) } **fun** main(args: Array<String>){  **val** reader = Scanner(System.*`in`*)  **for**(i **in** 1..5){  **val** x = reader.nextDouble()  *powerA3*(x);  } } | | | |
| VB.NET | | | |
| Sub PowerA3(ByRef x As Integer, ByVal a As Integer)  x = a \* a \* a  End Sub  Sub Main()  Dim a, b As Integer  For i As Integer = 1 To 5  a = Convert.ToInt32(Console.ReadLine())  PowerA3(b, a)  Console.WriteLine(b)  Next  Console.ReadKey()  End Sub | | | |

1. Описать процедуру , вычисляющую *среднее арифметическое* и *среднее геометрическое*  двух положительных чисел *X* и *Y* (*X* и *Y* — входные,   
   и — выходные параметры вещественного типа). С помощью этой процедуры найти среднее арифметическое и среднее геометрическое для пар (*A*, *B*), (*A*, *C*), (*A*, *D*), если даны *A*, *B*, *C*, *D*.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **procedure** Mean(X, Y: Real; **var** AMean, GMean: Real);  **begin**  AMean := (X + Y) / 2;  GMean := sqrt(X \* Y);  **end**;  **var**  i: Integer;  A, B, C, D, AMean, GMean: Real;  **begin**    Write('A,B,C,D: ');  Readln(A, B, C, D);    Mean(A, B, AMean, GMean);  Writeln(AMean, ' ', GMean);    Mean(A, C, AMean, GMean);  Writeln(AMean, ' ', GMean);    Mean(A, D, AMean, GMean);  Writeln(AMean, ' ', GMean);  **end**. | | #include <iostream>  #include <cmath>  using namespace std;  void Mean(float x, float y, float &a, float &g) {  a = (x + y) / 2;  g = sqrt(x\*y);  }  int main(){  float a, b, c, d, x, y;  cin >> a >> b >> c >> d;  Mean(a, b, x, y);  cout << x << " " << y << endl;  Mean(a, c, x, y);  cout << x << " " << y << endl;  Mean(a, d, x, y);  cout << x << " " << y << endl;  system("pause");  return 0;  } |
| Python | C# | |
| import math  def Mean(x,y):  a=(x+y)/2  g=math.sqrt(x\*y)  return(a,g)  for i in range(0,5):  a=float(input("a"))  b=float(input("b"))  print(Mean(a,b)) | static void Mean(double x, double y, out double a, out double g)  {  a=(x+y)/2;  g=Math.Sqrt(x\*y);  }  static void Main(string[] args)  {  double a,b,c,d,x,y;  a=Convert.ToSingle(Console.ReadLine());  b=Convert.ToSingle(Console.ReadLine());  c=Convert.ToSingle(Console.ReadLine());  d=Convert.ToSingle(Console.ReadLine());  Mean(a,b, out x, out y);  Console.WriteLine("AMean={0}  GMean={1}", x,y);  Mean(a, c, out x, out y);  Console.WriteLine("AMean={0}  GMean={1}", x, y);  Mean(a, d, out x, out y);  Console.WriteLine("AMean={0}  GMean={1}", x, y);  Console.ReadKey();  } | |
| Go | | |
| package main  import (      "fmt"      "math"  )  func Mean(x,y float64) (float64,float64) {  var a=(x+y)/2;  var g=math.Sqrt(x\*y)  return a,g  }  func main() {  var a,b float64  for i:=0; i<5; i++ {  fmt.Print("a, b->")  fmt.Scan(&a,&b)  fmt.Println(  Mean(a,b))  }  } | | |

1. Описать процедуру DigitCountSum(*K*, *C*, *S*), находящую количество *C* цифр целого положительного числа *K*, а также их сумму *S* (*K* — входной, *C* и *S* — выходные параметры целого типа). С помощью этой процедуры найти количество и сумму цифр для каждого из пяти данных целых чисел.

| PascalABC.NET | C++ |
| --- | --- |
| **procedure** DigitCountSum(K: Integer; **var** C, S: Integer);  **var**  TempI: Integer;  **begin**  C := 0;  S := 0;  **while** (k <> 0) **do**  **begin**  S := S + (K **mod** 10);  C := C + 1;  K := K **div** 10;  **end**;  **end**;  **var**  i, K, C, S: Integer;  **begin**  **for** i := 1 **to** 5 **do**  **begin**  Write('K: ');  Readln(K);  DigitCountSum(K, C, S);  Writeln(C, ' ', S);  **end**;  **end**. | #include <iostream>  using namespace std;  void DigitCountSum(int k, int &c, int &S) {  c = S = 0;  while (k != 0) {  S = S + k / 10;  c++;  k = k / 10;  }  }  int main(){  int i,k, c, S;  for (i = 0; i < 5; i++) {  cin >> k;  DigitCountSum(k, c, S);  cout << "c="<<c << "  S=" << S << endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import "fmt"  func digitcountsum(x int) (int, int) {  var s=0  var k=0  for x!=0{  s=s+x%10  x=x/10  k++;  }  return k,s  }  func main() {  var a int  for i:=0; i<5; i++ {  fmt.Print("a->")  fmt.Scan(&a)  fmt.Println(digitcountsum(a))  }  } | |

1. Описать процедуру Swap(*X*, *Y*), меняющую содержимое переменных *X* и *Y* (*X* и *Y* — вещественные параметры, являющиеся одновременно входными и выходными). С ее помощью для данных переменных *A*, *B*, *C*, *D* последовательно поменять содержимое следующих пар: *A* и *B*, *C* и *D*, *B* и *C* и вывести новые значения *A*, *B*, *C*, *D*.

| PascalABC.NET | C++ |
| --- | --- |
| **procedure** Swap(**var** X,Y: Real);  **begin**  (X,Y):=(Y,X);  **end**;    **var**  A,B,C,D:Real;    **begin**  Write('A,B,C,D :');  Readln(A,B,C,D);  Swap(A,B);  Swap(C,D);  Swap(B,C);  Writeln(A,' ',B,' ',C,' ',D);  **end**. | #include <iostream>  using namespace std;  void swap(int &a, int &b) {  int h = a;  a = b;  b = h;  }  int main(){  int a, b, c, d;  cin >> a >> b >> c >> d;  swap(a, b);  swap(c, d);  swap(b, c);  cout << a << " " << b << " "  << c << " " << d << endl;  system("pause");  return 0;  } |
| Go | |
| package main  import "fmt"  func swap(x,y int) (int, int) {  return y,x  }  func main() {  var a,b,c,d int  fmt.Print("a,b,c,d->")  fmt.Scan(&a,&b,&c,&d)  fmt.Println(swap(a,b))  fmt.Println(swap(c,d))  fmt.Println(swap(b,c))  } | |
| Kotlin | |
| import java.util.Scanner fun swap(a: Float, b:Float): Unit {  println("b=$a a=$b") } fun main(args: Array<String>){  val reader = Scanner(System.`in`)  var a=reader.nextFloat()  var b=reader.nextFloat()  var c=reader.nextFloat()  var d=reader.nextFloat()  swap(a,b)  swap(c,d)  swap(b,c) } | |

## **Функции с числовыми параметрами**

1. Описать функцию Sign(*X*) целого типа, возвращающую для вещественного числа *X* следующие значения:

–1,    если *X* < 0;        0,    если *X* = 0;        1,    если *X* > 0.

С помощью этой функции найти значение выражения Sign(*A*) + Sign(*B*) для данных вещественных чисел *A* и *B*.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **function** Sign(X: Real): Integer;  **begin**  **if** X<0 **then** Sign:=-1;  **if** X=0 **then** Sign:=0;  **if** X>0 **then** Sign:=1;  **end**;  **var**  A, B: Real;  **begin**  Write('A,B :');  Readln(A, B);  Writeln(Sign(A) + Sign(B));  **end**. | | #include <iostream>  using namespace std;  int sign(float x) {  if (x<0)  return -1;  if (x == 0)  return 0;  if (x>0)  return 1;  }  int main(){  float a, b;  cin >> a >> b;  int c = sign(a) + sign(b);  cout << c<<endl;  system("pause");  return 0;  } |
| Python | C# | |
| def sign(x):  if x<0:  return -1  elif x>0:  return 1  else:  return 0  a=float(input("a"))  b=float(input("b"))  S=sign(a)+sign(b)  print(S) | package main  import "fmt"  func sign(x float64) (int) {  if x<0{  return -1  } else if x>0{  return 1  } else{  return 0  }  }  func main() {  var a, b float64  fmt.Print("a,b->")  fmt.Scan(&a,&b)  var c=sign(a)+sign(b)  fmt.Println(c)  } | |
| Kotlin | | |
| import java.util.Scanner  fun sign(a: Float) :Int{  if (a<0)  return -1  else if (a>0)  return 1  else  return 0  }  fun main(args: Array<String>) {  val reader = Scanner(System.`in`)  var a = reader.nextFloat()  var b = reader.nextFloat()  var c = sign(a) + sign(b);  println(c)  } | | |
| VB.NET | | |
| Function Sign(x As Single) As Integer  If x > 0 Then  Return 1  ElseIf x < 0 Then  Return -1  Else  Return 0  End If  End Function  Sub Main()  Dim a, b As Single  Dim m As Integer  a = Convert.ToSingle(Console.ReadLine())  b = Convert.ToSingle(Console.ReadLine())  m = Sign(a) + Sign(b)  Console.WriteLine(m)  Console.ReadKey()  End Sub | | |

1. Описать функцию SumRange(*A*, *B*) целого типа, находящую сумму всех целых чисел от *A* до *B* включительно (*A* и *B* — целые). Если *A* > *B*, то функция возвращает 0. С помощью этой функции найти суммы чисел от *A* до *B* и от *B* до *C*, если даны числа *A*, *B*, *C*.

| PascalABC.NET | C++ |
| --- | --- |
| **function** SumRange(A, B: Integer): Integer;  **var**  i, S: Integer;  **begin**  S := 0;  **for** i := A **to** B **do**  **begin**  S := S + i;  **end**;  SumRange := S;  **end**;  **var**  A, B, C: Integer;  **begin**  Write('A,B,C :');  Readln(A, B, C);  Writeln(SumRange(A, B));  Writeln(SumRange(B, C));  **end**. | #include <iostream>  using namespace std;  int sumrange(int a, int b) {  int sum = 0;  for (; a <= b; ++a)  sum += a;  return sum;  }  int main(){  int a, b, c, S, S1;  cin >> a >> b>>c;  S = sumrange(a, b);  S1 = sumrange(b, c);  cout <<S<<" "<<S1<<endl;  system("pause");  return 0;  } |
| Go | |
| package main  import "fmt"  func sum(a,b int) (int) {  var s=0  for ;a<=b; a++{  s+=a  }  return s  }  func main() {  var a, b,c int  fmt.Print("a,b,c->")  fmt.Scan(&a,&b, &c)  fmt.Println(sum(a,b))  fmt.Println(sum(b,c))  } | |

1. Описать функцию IsPrime(*N*) логического типа, возвращающую True, если целый параметр *N* (> 1) является простым числом, и False в противном случае (число, большее 1, называется *простым*, если оно не имеет положительных делителей, кроме 1 и самого себя). Дан набор из 10 целых чисел, больших 1. С помощью функции IsPrime найти количество простых чисел в данном наборе.

| PascalABC.NET | C++ |
| --- | --- |
| **function** IsPrime(N: Integer): Boolean;  **var**  i: integer;  **begin**  i := 1;  IsPrime := True;  **while**(i<=trunc(sqrt(N)))**do**  **begin**  Inc(i);  **if** N **mod** i = 0 **then**  **begin**  I := N;  IsPrime := False;  **end**;  **end**;  **end**;  **var**  i, N, Res: Integer;  **begin**  Res := 0;  **for** i := 1 **to** 10 **do**  **begin**  Write('N :');  Readln(N);    **if** IsPrime(N) **then**  Inc(Res);  **end**;  Writeln(Res);  **end**. | #include <iostream>  #include <cmath>  using namespace std;  int isprime(int n) {  int i, p = 1;  for (i=2;i<=(int)sqrt(n);i++)  if (n%i == 0) {  p = 0;  break;  }  return p;  }  int main(){  int i, n, res = 0;  for (i = 1; i <= 10; ++i) {  cin >> n;  res += isprime(n);  }  cout << res << endl;  system("pause");  return 0;  } |
| Go | |
| package main  import "fmt"  func isprime(a int) (bool) {  var y bool=true  for i:=2; i< a/2; i++{  if (a%i==0){  y=false  break  }  }  return y  }  func main() {  var a,k int=0,0  fmt.Print("a->")  for i:=1; i<=10; i++{  fmt.Scan(&a)  if (isprime(a)){  k++  }  }  fmt.Println(k)  } | |

## **Дополнительные задания на процедуры и функции**

1. Описать функцию Exp1(*x*, ) вещественного типа (параметры *x*, ε — вещественные, > 0), находящую приближенное значение функции exp(*x*):

exp(*x*) = 1 + *x* + *x*2/(2!) + *x*3/(3!) + … + *xn*/(*n*!) + …

(*n*! = 1·2·…·*n*). В сумме учитывать все слагаемые, большие ε. С помощью Exp1 найти приближенное значение экспоненты для данного *x* при шести данных ε.

| PascalABC.NET | C++ |
| --- | --- |
| **function** Exp1(x,e:Real):Real;  **var**  i: integer;  xi, ni: Real;  TempRez, Rez: Real;  **begin**  xi := 1;  i := 0;  ni := 1;  Rez := 0;  TempRez := 1;  **while** (TempRez > e) **do**  **begin**  Rez := Rez + TempRez;  inc(i);  xi := xi \* x;  ni := ni \* i;  TempRez := xi / ni;  **end**;  Exp1 := Rez;  **end**;  **var**  X, E: real;  i: integer;  **begin**  Write('X :');  Readln(X);  **for** i := 1 **to** 6 **do**  **begin**  Write('E :');  Readln(E);  Writeln(Exp1(X, E));  **end**;  **end**. | #include <iostream>  using namespace std;  float exp1(float x, float e) {  float xi=1,ni=1,rez=0,h=1;  int i = 0;  while (temprez>e) {  rez += temprez;  ++i;  xi \*= x;  ni \*= i;  temprez = xi / ni;  }  return rez;  }  int main(){  float x, e;  cin >> x;  for (int i = 1; i <= 6; ++i) {  cin >> e;  cout << exp1(x, e) << endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math"  )  func exp1(x,e float64) float64 {  var s,p,t, i float64=1,1,1,1  for t>=e{  p=p\*i;  t=math.Pow(x,i)/p  s=s+t  i++;  }  return s  }  func main() {  var x,e float64  fmt.Print("x")  fmt.Scan(&x)  for i := 0; i < 10; i++{  fmt.Scan(&e)  fmt.Println(exp1(x,e))  }  } | |

1. Описать процедуру TimeToHMS(*T*, *H*, *M*, *S*), определяющую по времени *T* (в секундах) содержащееся в нем количество часов *H*, минут *M* и секунд *S* (*T* — входной, *H*, *M* и *S* — выходные параметры целого типа). Используя эту процедуру, найти количество часов, минут и секунд для пяти данных отрезков времени *T*1, *T*2, …, *T*5.

| PascalABC.NET | C++ |
| --- | --- |
| **procedure** TimeToHMS(T: integer; **var** H,M,S:integer);  **begin**  H := T **div** 3600;  T := T - H \* 3600;  M := T **div** 60;  S := T - M \* 60  **end**;  **var**  T, H, M, S: integer;  i: byte;  **begin**  **for** i := 1 **to** 5 **do**  **begin**  write('T', i, ' = ');  readln(T);  **if** T >= 0 **then**  **begin**  TimeToHMS(T, H, M, S);  writeln(H,':',M,':',S);  **end**  **else**  writeln(Error!');  **end**  **end**. | #include <iostream>  using namespace std;  void TimeToHMS(int T, int &H, int  &M, int &S) {  H = T / 3600;  M = (T % 3600) / 60;  S = T % 60;  }  int main()  {  int T, H, M, S;  for (int i = 1; i <= 5; ++i) {  cin >> T;  TimeToHMS(T, H, M, S);  cout<<H<<":"<<M<<":"<<S<<endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import "fmt"  func TimeToHMS(T int) (int, int, int) {  var H = T /3600  var M = (T % 3600)/60  var S = T%60  return H,M,S  }  func main() {  var t,h,m,s int  for i := 0; i < 3; i++{  fmt.Print("t->")  fmt.Scan(&t)  h,m,s=TimeToHMS(t)  fmt.Printf("%v:%v:%v\n", h,m,s)  }  } | |

**Задачи для самостоятельной работы**

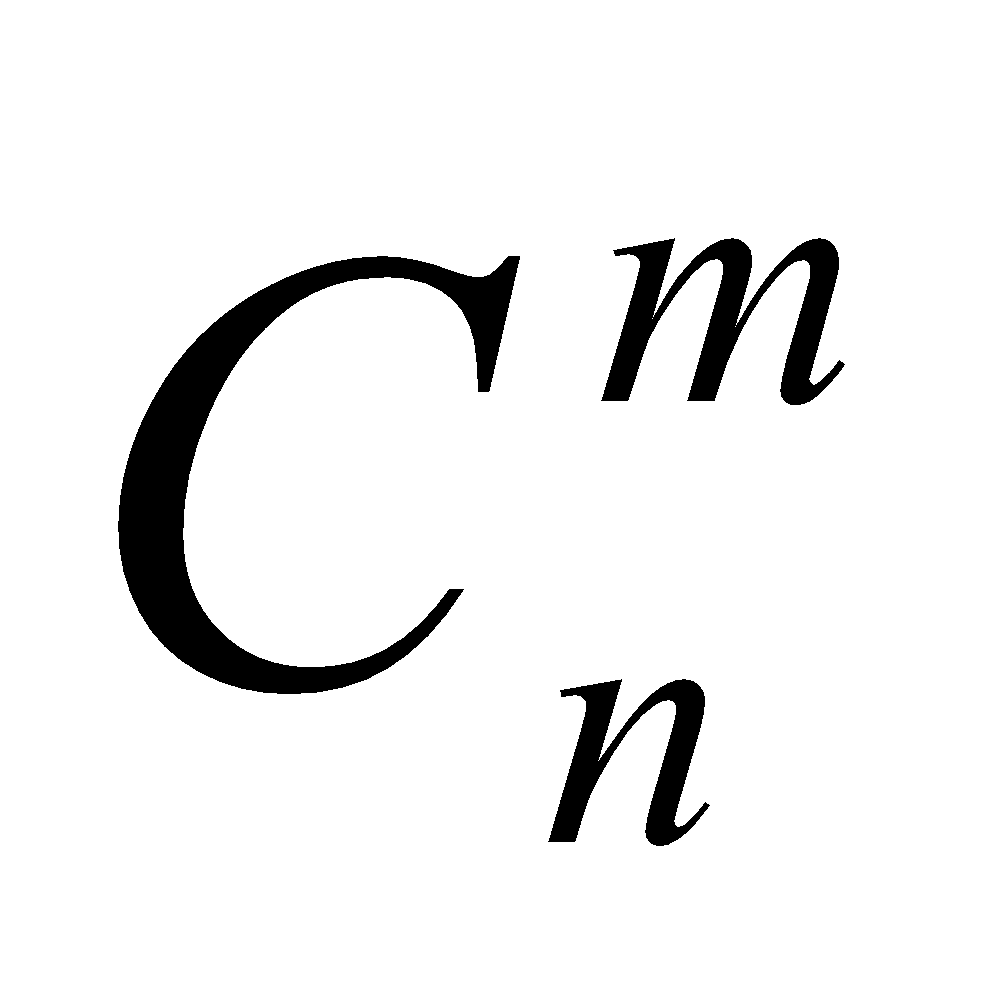
**Вариант 1**

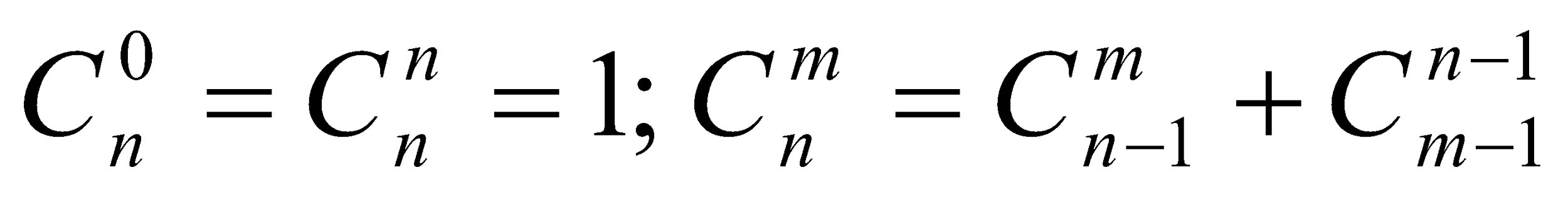
1. Дано натуральное число *N*. Составить процедуру (функцию) формирования массива, элементами которого являются цифры числа *N*.
2. Треугольник задан координатами своих вершин. Составить функцию вычисления его площади.
3. Найдите сумму цифр заданного натурального числа, используя рекурсивную подпрограмму.

**Вариант 2**

1. Составить процедуру (функцию), определяющую, в каком из данных двух чисел больше цифр.
2. Составить функцию нахождения наибольшего общего делителя и наименьшего общего кратного двух натуральных чисел (*НОК*(*А*, *В*) = *А\*Б*/ *НОД*(*А*, *В*).
3. Подсчитать количество цифр в заданном натуральном числе, используя рекурсивную подпрограмму.

**Вариант 3**

1. Заменить данное натуральное число на число, которое получается из исходного записью его цифр в обратном порядке. Например, дано число 156, нужно получить 651.
2. Составить функцию нахождения наибольшего общего делителя четырех натуральных чисел.
3. Описать функцию С(m,n), где 0≤ m≤ n, для вычисления биномиального коэффициента по следующей формуле:

 при 0< *m*< *n*

используя рекурсивную подпрограмму

**Вариант 4**

1. Даны натуральные числа *K* и *N*. Составить процедуру (функцию) формирования массива *A*, элементами которого являются числа, сумма цифр которых равна *K* и которые не больше *N*.
2. Составить функцию нахождения наименьшего общего кратного трех натуральных чисел.
3. Описать рекурсивную функцию Root(*f*, *b*, *ε*), которая методом деления отрезка пополам находит с точностью ε корень уравнения f(x) = 0 на отрезке [*а*, *b*] (считать, что *ε* > 0, *а* < *b*, *f*(*a*} • *f*(*b*) < 0 и *f*(*x*)- непрерывная и монотонная на отрезке [*а*, *b*]).

**Вариант 5**

1. Даны три квадратных матрицы *A*, *B*, *С* *n*-го порядка. Вывести на печать ту из них, норма которой наименьшая.

Пояснение. Нормой матрицы считать максимум из абсолютных величин ее элементов.

1. Написать функцию нахождения суммы большего и меньшего из 3 чисел.
2. Описать функцию *min*(*x*) для определения минимального элемента линейного массива х, введя вспомогательную рекурсивную функцию *min*1(*k*), находящую минимум среди последних элементов массива *х*, начиная с *k*-го.

**Вариант 6**

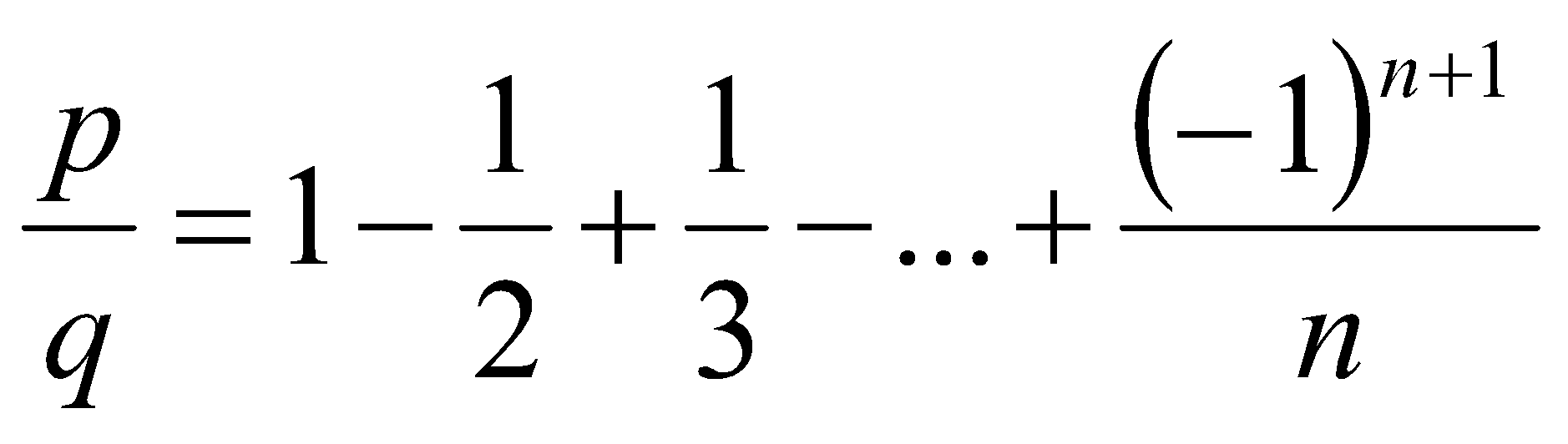
1. Два натуральных числа называются «дружественными», если каждое из них равно сумме всех делителей (кроме его самого) другого (например, числа 220 и 284). Найти все пары «дружественных чисел», которые не больше данного числа *N*.
2. Вычислить площадь правильного шестиугольника со стороной *a*, используя функцию вычисления площади треугольника.
3. Описать рекурсивную логическую функцию Simm(*S*, i, j), проверяющую, является ли симметричной часть строки *S*, начинающаяся *i*-м и кончающаяся *j*-м ее элементами.

**Вариант 7**

1. Два простых числа называются «близнецами», если они отличаются друг от друга на 2 (например, 41 и 43). Напечатать все пары «близнецов» из отрезка [*n*, 2*n*], где *n* — заданное натуральное число больше 2.
2. На плоскости заданы своими координатами *n* точек. Составить функцию, определяющую между какими из пар точек самое большое расстояние. Указание: координаты точек занести в массив.
3. Составить программу вычисления наибольшего общего делителя двух натуральных чисел, используя рекурсивную подпрограмму.

**Вариант 8**

1. Написать процедуру (функцию) вычисления суммы ряда



для заданного числа *n*. Дробь p/q должна быть несократимой (*р*, *q* — натуральные).

Указание: основой вычислений должна быть процедура сложения двух простых дробей.

1. Составить функцию, которая в массив- *A*[*N*] находит второе по величине число (т.e. вывести на печать число, которое меньше максимального элемента массива, но больше всех других элементов).
2. Составить программу нахождения числа, которое образуется из данного натурального числа при записи его цифр в обратном порядке, используя рекурсивную подпрограмму. Например, для числа 1234 получаем ответ 4321.

**Вариант 9**

1. Написать процедуру (функцию) вычисления суммы 1 + 1/2 + 1/3 + …+1/*n* для заданного числа *n*. Результат представить в виде несократимой дроби *p*/*q* (*p*, *q* — натуральные).
2. Составить функцию, проверяющую, являются ли данные три числа взаимно простыми.
3. Составить программу перевода данного натурального числа в *р*-ичную систему счисления (2 ≤ р≤ 9) ,используя рекурсивную подпрограмму.

**Вариант 10**

1. Натуральное число, в записи которого *n* цифр, называется числом Амстронга, если сумма его цифр, возведенная в степень *n*, равна самому числу. Найти все числа Амстронга от 1 до *k*.
2. Написать функцию вычисления суммы факториалов всех нечетных чисел от 1 до 9.
3. Дана символьная строка, представляющая собой запись натурального числа в *р*-ичной системе счисления (2 ≤ *р* ≤ 9). Составить программу перевода этого числа в десятичную систему счисления , используя рекурсивную подпрограмму.

**Вариант 11**

1. Написать процедуру (функцию), которая находит и выводит на печать все четырехзначные числа вида *abcd*, для которых выполняется: а) *а*, *b*, *с*, *d* — разные цифры; б) *ab* - *cd* = *а* + *b* + *с* + *d*.
2. Даны две дроби *A*/*B* и *C*/*D*. (*А*, *В*, *С*, *D* - натуральные числа). Составить функцию деления дроби на дробь. Ответ должен быть несократимой дробью.
3. Составить программу вычисления суммы:

1! +2! + З! + … +*n*! (*n* ≤ 20).

**Примечание**. Тип результата значения функции - LongInt.

**Вариант 12**

1. Найти все простые натуральные числа, не превосходящие *n*, двоичная запись которых представляет собой палиндром, т.е. читается одинаково слева направо и справа налево.
2. Задан массив *D* из 6 элементов. Определить следующие суммы: *D*[1] + *D*[2] + *D*[3]; *D*[3] + *D*[4] + *D*[5]; *D*[4] + *D*[5] + *D*[6]. Пояснение. Составить функцию вычисления суммы трех последовательно расположенных элементов массива с номерами от *k* до *m* .
3. Составить программу вычисления суммы:

2! + 4! + … + *n*! (*n* ≤ 20, *n* - четное).

**Примечание**: Тип результата значения функции — LongInt.

**Вариант 13**

1. Найти все натуральные четырехзначные числа, цифры в которых образуют строго возрастающую последовательность (например, 1234, 5789).
2. Даны две дроби *A*/*B* и *C*/*D* (*А*, *В*, *С*, *D* — натуральные числа). Составить функцию умножения дроби на дробь. Ответ должен быть несократимой дробью.
3. Дано *n* различных натуральных чисел. Напечатать все перестановки этих чисел.

**Вариант 14**

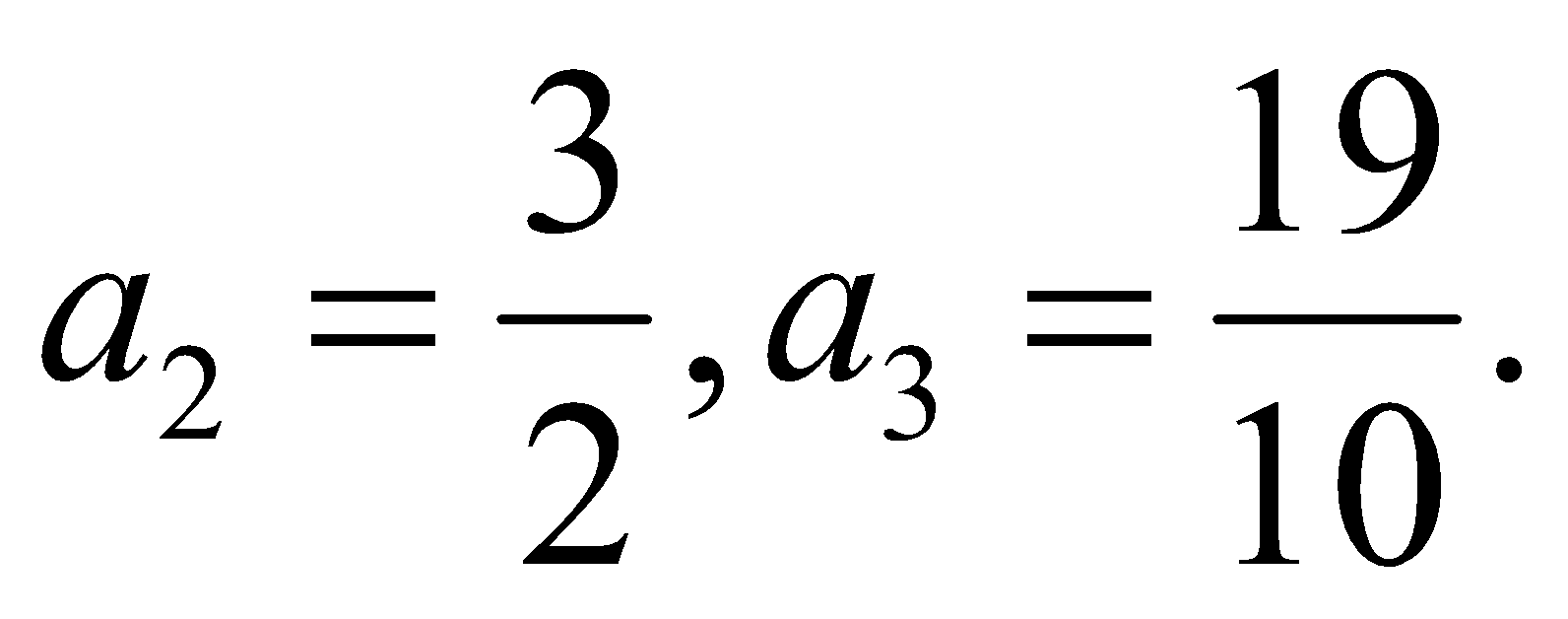
1. Найти все натуральные числа, не превосходящие заданного *n*, которые делятся на каждую из своих цифр.
2. Даны две дроби *A*/*B* и *C*/*D* (*А*, *В*, *С*, *D* — натуральные числа). Составить функцию вычитания из первой дроби второй. Ответ должен быть несократимой дробью.
3. Ежедневно Незнайка учит половину от суммы выученных за два предыдущих дня иностранных слов и еще два слова. Знайка считает, что силы Незнайки иссякнут, когда нужно будет выучить 50 слов в день. Написать программу, определяющую, через сколько дней иссякнут силы у Незнайки, если в первые два дня он выучил по одному слову.

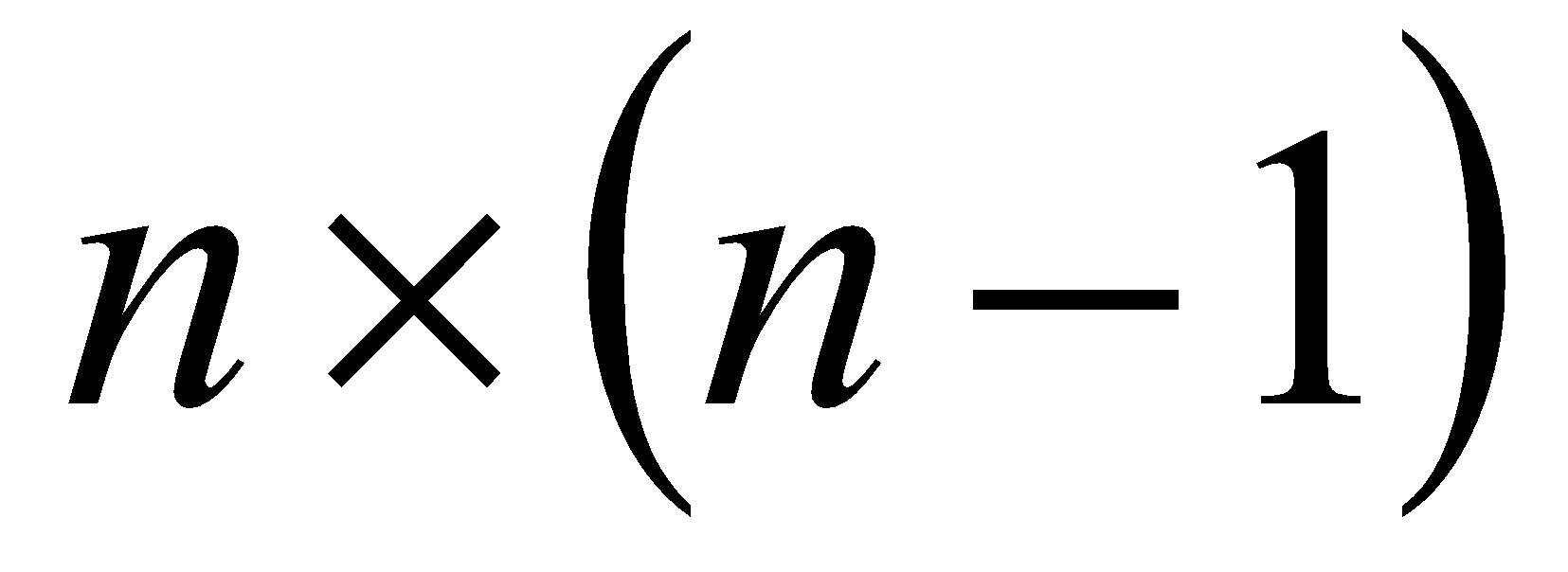
**Вариант 15**

1. Составить процедуру (функцию) для нахождения чисел из интервала [*М*; *N*], имеющих наибольшее количество делителей.
2. Даны две дроби *A*/*B* и *C*/*D* (*А*, *В*, *С*, *D* — натуральные числа). Составить функцию сложения этих дробей. Ответ должен быть несократимой дробью.
3. Татьяна Ларина, читая очередной французский роман, подсчитала сумму номеров прочитанных страниц. Обозначим эту сумму Q. Написать программу, определяющую номер последней прочитанной страницы.

**Вариант 16**

1. Для последовательности *а*1 = 1, *an+1* = *an* + 1/(1 + *an*) составить процедуру (функцию) печати *k*-го члена в виде обыкновенной несократимой дроби. Например,



1. На плоскости заданы своими координатами *n* точек. Создать массив размерности , элементами которого являются расстояния от каждой из точек до *n* - 1 других.
2. Царевна-лягушка съедает ежедневно на 20% комаров больше, чем в предыдущий день, и еще два комара. Написать программу, определяющую через сколько дней количество съеденных комаров превысит 100, если в первый день было съедено 12 комаров.

**Вариант 17**

1. Дано натуральное число *n*. Выяснить, можно ли представить n в виде произведения трех последовательных натуральных чисел.
2. Даны 4 числа *X*, *Y*, *Z*, *Т* — длины сторон четырехугольника. Вычислить его площадь, если угол между сторонами длиной *Х* и *Y* – прямой.
3. На каждом следующем дне рождения Винни Пух съедает столько же пищи, что и на двух предыдущих. На двух первых днях рождения у Пятачка и Кролика он съел по 100 г пищи. Написать программу, определяющую, сколько килограммов пищи съест Винни Пух на пятнадцатом дне рождения.

**Вариант 18**

1. Имеется часть катушки с автобусными билетами. Номер билета шестизначный. Составить программу, определяющую количество счастливых билетов на катушке, если меньший номер билета — *N*, больший — *М* (билет является счастливым, если сумма первых трех его цифр равна сумме последних трех).
2. Сформировать массив *X*[*N*], *n*-й член которого определяется формулой *X*(*N*) = 1/*N*!.
3. Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько клеток будет через 3, 6, 9, 12, ..., 24 часа.

**Вариант 19**

1. Написать процедуру (функцию), определяющую сумму трехзначных чисел, содержащих только нечетные цифры. Определить также, сколько четных цифр в найденной сумме.
2. Составить функцию вычисления суммы факториалов всех четных чисел от *m* до *n.*
3. Начав тренировки, спортсмен в первый день пробежал 10 км. Каждый день он увеличивал дневную норму на 10% от нормы предыдущего дня. Какой суммарный путь пробежит спортсмен за 7 дней?

**Вариант 20**

1. Из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т.д. Через сколько таких действий получится нуль?
2. Заменить отрицательные элементы линейного массива их модулями, не пользуясь стандартной функцией вычисления модуля. Подсчитать количество произведенных замен.
3. Составить программу вычисления суммы:

1! + 3! + … + *5*! (*n* ≤ 20, *n* - нечетное).

**Примечание**: Тип результата значения функции — LongInt.

**Вариант** 21

1. Составить процедуру (функцию) разложения данного натурального числа на простые множители. Например, 200 = 23 • 52.
2. Дан массив *A*[*N*]. Сформировать массив *В*[.*М*], элементами которого являются большие из двух рядом стоящих в массиве *A* чисел. (Например, массив *А* состоит из элементов 1, 3, 5, -2, 0, 4, 0. Элементами массива *В* будут 3, 5, 4.)
3. Составить программу вычисления суммы:

2! - 4! +6!- … + *n*! (*n* ≤ 20, *n* - четное).

**Примечание**: Тип результата значения функции — LongInt.

**Вариант 22**

1. Дано натуральное число n. Найти все меньшие n числа Мерсена. (Простое число называется числом Мерсена, если оно может быть представлено в виде 2p— 1, где р — тоже простое число. Например, 31 = 25 — 1 — число Мерсена.)
2. Дан массив *A*[*N*] (*N* — четное число). Сформировать массив *В*[*М*], элементами которого являются средние арифметические соседних пар рядом стоящих в массиве *А* чисел. Например, массив *A* состоит из элементов 1, 3, 5, -2, 0, 4, 0, 3. Элементами массива *В* будут 2, 1.5, 2, 1.5.
3. Составить программу вычисления суммы:

1! - 3! +5!- … + (2*n-1)*! (*n* ≤ 20, *n* - четное).

**Примечание**: Тип результата значения функции — LongInt.

**Вариант 23**

1. Дано четное число *n* > 2. Проверить для него гипотезу Гольдбаха: каждое четное n представляется в виде суммы двух простых чисел.
2. Дано простое число. Составить функцию, которая будет находить следующее за ним простое число.
3. Перемножая большие числа, можно быстро получить переполнение. Поэтому, для того чтобы напечатать произведение, превышающее наибольшее допустимое для данного целого типа (integer или longint) числа, надо применить искусственные средства.

**Вариант 24**

1. Даны координаты вершин двух треугольников. Написать программу, позволяющую определить, площадь которого из них больше, используя для этого подпрограмму вычисления площади треугольника.
2. Описать функцию Sin1(*x*, ) вещественного типа (параметры *x*, — вещественные, > 0), находящую приближенное значение функции sin(*x*):

sin(*x*) = *x* – *x*3/(3!) + *x*5/(5!) – … + (–1)*n*·*x*2·*n*+1/((2·*n*+1)!) + … .

В сумме учитывать все слагаемые, модуль которых больше . С помощью Sin1 найти приближенное значение синуса для данного *x* при шести данных .

1. Составить функцию для нахождения наименьшего нечетного натурального делителя *k* (*k* ≠ 1) любого заданного натурального числа *n*.

**Вариант 25**

1. Составить функцию для нахождения наименьшего нечетного натурального делителя *k* (*k* ≠ 1) любого заданного натурального числа *n*.
2. Описать функцию IsPrime(N) логического типа, возвращающую True, если целый параметр N (> 1) является простым числом, и False в противном случае (число, большее 1, называется простым, если оно не имеет положительных делителей, кроме 1 и самого себя). Дан набор из 10 целых чисел, больших 1. С помощью функции IsPrime найти количество простых чисел в данном наборе.
3. Описать рекурсивную функцию Palindrom(S) логического типа, возвращающую True, если строка S является палиндромом (то есть читается одинаково слева направо и справа налево), и False в противном случае. Оператор цикла в теле функции не использовать. Вывести значения функции Palindrom для пяти данных строк.

# **Одномерные массивы: группа Array**

Условие вида «дан массив размера *N*» означает, что вначале дается *фактический размер* массива (целое число *N*), а затем приводятся все его элементы. Если в задании явно не указывается, какие значения может принимать размер исходного массива, то предполагается, что размер может изменяться в пределах от 2 до 10. Порядковый номер начального элемента массива считается равным 1.

Если в задании, связанном с созданием (преобразованием) массива, не описан результирующий набор данных, то предполагается, что этим набором является созданный (преобразованный) массив, и необходимо вывести все его элементы в порядке возрастания их индексов.

## **Формирование массива и вывод его элементов**

В заданиях на формирование массива предполагается, что размер результирующего массива не превосходит 10.

1. Дано целое число *N* (> 0). Сформировать и вывести целочисленный массив размера *N*, содержащий *N* первых положительных нечетных чисел: 1, 3, 5, … .

| PascalABC.NET | | C++ | | |
| --- | --- | --- | --- | --- |
| **var**  a: **array of** integer;  N, i: Integer;  **begin**  Write('N:');  Readln(N);  a:=**new** integer[N+1];  **for** i := 1 **to** N **do**  a[i] := i \* 2 - 1;  print(a);  **end**. | | #include <iostream>  using namespace std;  int main(){  int \*a,i,n;  cin >> n;  a = new int[n];  for (i = 0; i<n; ++i)  a[i] = 2\*i+1;  for (i = 0; i<n; ++i)  cout<<a[i]<<" ";  system("pause");  return 0;  } | | |
| Python | C# | | | |
| a=[]  n=int(input(""))  for i in range (1,n):  a.append(2\*i-1)  print(a) | static void Main(string[] args){  int n, I;  n=Convert.ToInt32(Console.ReadLine());  int []x=new Int32[n];  for (i=0; i<n; i++){  x[i]=2\*i+1;  Console.Write(x[i]+" ");  }  Console.ReadKey();  } | | | |
| Java | | | | Go |
| public static void main(String[] args){  Scanner sc=new Scanner(System.in);  int n, i;  n=sc.nextInt();  int [] x=new int[n];  for (i=0; i<n; i++){  x[i]=2\*i+1;  System.out.print(x[i]+" ");  }  } | | | | package main  import "fmt"  func main() {  var n int  fmt.Print("n->")  fmt.Scan(&n)  var a [] int  a=make([]int, n)  for i := 0; i < n; i++{  a[i]=2\*i+1  }  fmt.Println(a)  } |
| Kotlin | | | | |
| import java.util.Scanner fun main(args: Array<String>) {  val reader=Scanner(System.`in`)  var n=reader.nextInt()  var x = Array<Int>(n){0}  for (i in 0..n-1){  x[i]=2\*i-1  }  for(i in 0..n-1) {  println(x[i])  } } | | | import java.util.Scanner fun main(args: Array<String>) {  val reader=Scanner(System.`in`)  var n=reader.nextInt()  val x = Array(n,{i->2\*i+1})  for(i in 0..n-1) {  println(x[i])  } } | |
| VB.NET | | | | |
| Sub Main()  Dim a() As Integer  Dim i, n As Integer  n = Convert.ToInt32(Console.ReadLine())  a = New Integer(n) {}  For i = 0 To n - 1  a(i) = 2 \* i + 1  Console.Write("{0} ", a(i))  Next  Console.ReadKey()  End Sub | | | | |

1. Дано целое число *N* (> 1), а также первый член *A* и разность *D* *арифметической прогрессии*. Сформировать и вывести массив размера *N*, содержащий *N* первых членов данной прогрессии:

*A*,    *A* + *D*,    *A* + 2·*D*,    *A* + 3·*D*,    … .

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Var**  a: **array of** integer;  D, N, i: Integer;  **begin**  Write('N:');  Read(N);  a:= **new** integer[N + 1];  Write('A1,D:');  Readln(a[1], D);  **for** i := 2 **to** N **do**  a[i]:=a[1]+(i-1)\*D;  **for** i := 1 **to** N **do**  writeln(i,':',a[i]);  **end**. | | #include <iostream>  using namespace std;  int main(){  int \*x,i,n, a,d;  cin >> n>>a>>d;  x= new int[n];  for (i = 0; i<n; ++i)  x[i] = a+i\*d;  for (i = 0; i<n; ++i)  cout<<x[i]<<" ";  system("pause");  return 0;  } |
| Python | Go | |
| N = int(input("N:"))  a=[]  a.append(int(input("A1:")))  D = int(input("D:"))    for i in range(1,N-1):      a.append(a[0]+(i)\*D)    for i in range(N-1):      print(a[i], end=' ') | package main  import "fmt"  func main() {  var n,a,d int  fmt.Print("n,a,d->")  fmt.Scan(&n,&a,&d)  var x [] int  x=make([]int, n)  for i := 0; i < n; i++{  x[i]=a+i\*d  }  fmt.Println(x)  } | |

1. Дан массив размера *N*. Вывести его элементы в обратном порядке.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  a: **array of** integer;  N, i: Integer;  **begin**  Write('N:');  Readln(N);  a := **new** integer[N + 1];  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  **for** i := N **downto** 1 **do**  writeln(i, ' : ', a[i]);  **end**. | | #include <iostream>  using namespace std;  int main(){  int \*x,i,n;  cin >> n;  x= new int[n];  for (i = 0; i < n; ++i) {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  for (i = n-1; i>=0; i--)  cout<<x[i]<<" ";  system("pause");  return 0;  } | |
| Python | C# | | |
| n=int(input("n->"))  x=[]  for i in range(0,n):  x.append(int(input("x")));  for i in range(n-1,-1,-1):  print(x[i]) | Вариант 1.  static void Main(string[] args)  {  int n=Convert.ToInt32(Console.ReadLine());  int []x=new Int32[n];  Random rnd=new Random();  for (int i=0; i<n; i++){  x[i] = rnd.Next(30);  Console.Write(x[i]+" ");  }  Console.WriteLine();  for (int i= n - 1; i>= 0; i--)  Console.Write(x[i]+" ");  Console.ReadKey();  }  Вариант 2.  static void Main(string[] args)  {  int n=Convert.ToInt32(Console.ReadLine());  int []x=new Int32[n];  for (int i=0; i<n; i++)  x[i] = Convert.ToInt32(Console.ReadLine());  Console.WriteLine();  for (int i= n - 1; i>= 0; i--)  Console.Write(x[i]+" ");  Console.ReadKey();  } | | |
| Java | | | Go |
| Вариант 1.  public static void main(String[] args){  Scanner sc=new Scanner(System.in);  int n, i;  n=sc.nextInt();  int [] x=new int[n];  for (i=0; i<n; i++){  x[i]=sc.nextInt();  }  for (i=0; i<n; i++)  System.out.print(x[i]+" ");  System.out.println();  for (i=n-1; i>=0; i--)  System.out.print(x[i]+" ");  }  Вариант 2.  public static void main(String[] args){  Scanner sc=new Scanner(System.in);  int n, i;  n=sc.nextInt();  int [] x=new int[n];  Random rnd=new Random();  for (i=0; i<n; i++){  x[i]=rnd.nextInt(50);  System.out.print(x[i]+" ");  }  System.out.println();  for (i=n-1; i>=0; i--)  System.out.print(x[i]+" ");  } | | | package main  import "fmt"  func main() {  var n int  fmt.Print("n->")  fmt.Scan(&n)  var x=make([]int, n)  for i := 0; i < n; i++ {  fmt.Scan(&x[i])  }  fmt.Print("PrintArray \n")  for i := n-1; i >=0; i--{  fmt.Println(x[i])  }  } |

1. Дан массив *A* размера *N* и целое число *K* (). Вывести элементы массива с порядковыми номерами, кратными *K*: *AK*, *A*2·*K*, *A*3·*K*, …Условный оператор не использовать.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** integer;  N, i, K: Integer;  **begin**  (N, K) := readInteger2;  a := **new** integer[N + 1];  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  i := K;  **while** (i <= N) **do**  **begin**  writeln(i,':',a[i]);  i := i + K;  **end**;  **end**. | #include <iostream>  using namespace std;  int main(){  int \*x,i,n,k;  cin >> n>>k;  x= new int[n];  for (i = 0; i < n; ++i) {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  for (i = k - 1; i<n; i += k)  cout<<x[i]<<" ";  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var n,k int  fmt.Print("n,k->")  fmt.Scan(&n,&k)  var x=make([]int, n)  for i := 0; i < n; i++ {  x[i]=rand.Intn(50);  }  fmt.Println(x)  for i:=k-1; i<n; i+=k{  fmt.Println(x[i])  }  } | |

1. Дан массив A размера N. Вывести его элементы в следующем порядке:

| PascalABC.NET |
| --- |
| **Var**  a: **array of** integer;  N, i, P: word;  **begin**  readln(N);  a := **new** integer[N + 1];  **for** i := 1 **to** N **do**  read(a[i]);  writeln(' Результат: ');  P := N **div** 4;  **for** i := 1 **to** P **do**  write(a[2\*i-1],'',a[2\*i],'',a[N+2-2\*i],' ',a[N+1-2\*i],' ');  **if** N **mod** 4 > 0 **then**  write(a[2 \* P + 1], ' ');  **if** N **mod** 4 > 1 **then**  write(a[2 \* P + 2], ' ');  **if** N **mod** 4 > 2 **then**  write(a[N - 2 \* P])  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main(){  int \*x,i,j,n,p;  cin >> n;  x= new int[n];  for (i = 0; i < n; ++i) {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  p = n / 4;  for (i = 0; i<p; i++)  cout<<x[2\*i]<<" "<<x[2\*i+1]<<" "<<x[n-1-2\*i]<<" "<<x[n-2-2\*i]<<" ";  if (n % 4 == 1)  cout << x[(n / 2)] << " ";  else  for (j = 0; j < (n % 4); j++)  cout << x[(n / 2) - 1 + j] << " ";  system("pause");  return 0;  } |
| Go |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var n,p int  fmt.Print("n->")  fmt.Scan(&n)  p=n / 4  var x=make([]int, n)  for i := 0; i < n; i++ {  x[i]=rand.Intn(50);  }  fmt.Println(x)  for i:=0; i<p; i++{  fmt.Printf("%v %v %v %v",x[2\*i],x[2\*i+1],x[n-1-2\*i],  x[n-2-2\*i])  }  if (n%4==1){  fmt.Printf("%+v ", x[(n/2)])  }else{  for j:=0; j<(n%4); j++{  fmt.Printf("%+v ", x[(n/2)-1+j])  }  }  } |

## **Анализ элементов массива**

Для выполнения некоторых заданий из данного пункта не требуется одновременно хранить в памяти все исходные данные, поэтому использовать при их выполнении массивы, строго говоря, ***не нужно***. Однако применение массивов позволяет сделать алгоритмы решения этих заданий более простыми и наглядными. Задания из данного пункта можно дополнить заданиями из групп Series и Minmax, рассматривая их как задания на обработку массивов. С другой стороны, для тех заданий данного пункта, которые можно выполнить, не используя массивы, полезно реализовать и такие алгоритмы решения.

1. Дан массив *A* ненулевых целых чисел размера 10. Вывести значение первого из тех его элементов *AK*, которые удовлетворяют неравенству   
   *AK* < *A*10. Если таких элементов нет, то вывести 0.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array**[1..10] **of** integer;  k: Integer;  **begin**  **for** k := 1 **to** 10 **do**  **begin**  write(k, ' :');  readln(a[k]);  **end**;  k := 1;  **while**(A[k]>=A[10])**and**(k<10)**do**  inc(k);  **if** k = 10 **then**  writeln('0')  **else**  Writeln(a[k]);  **end**. | #include <iostream>  using namespace std;  int main(){  int x[11],i;  for (i = 1; i <=10; ++i) {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  i = 1;  while ((x[i]>=x[10])&&(i<10))  i++;  if(i == 10) cout << "0";  else cout << x[i] << endl;  system("pause");  return 0;  } |

1. Дан массив *A* размера *N*. Найти минимальный элемент из его элементов с четными номерами: *A*2, *A*4, *A*6, … .

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** integer;  N, i, Min: Integer;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  i := 2;  Min := a[2];  **while** (i <= N) **do**  **begin**  **if** a[i] < Min **then**  Min := a[i];  i := i + 2;  **end**;  Writeln(Min);  **end**. | #include <iostream>  using namespace std;  int main(){  int \*x,i, n;  cin >> n;  x = new int[n];  for (i = 0; i <n; ++i) {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  int min = x[0];  for (i = 1; i<n; i += 2) {  if (x[i]<min)  min = x[i];  }  cout << min<< endl;  system("pause");  return 0;  } |

1. Дан массив размера *N*. Найти номер его первого локального минимума (*локальный минимум* — это элемент, который меньше любого из своих соседей).

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a:**array of** integer;  N,i:Integer;  **begin**  Write('N: ');  Readln(N);  a:=**new** integer[N+1];  **for** i:=1 **to** N **do**  **begin**  write(i,' :');  readln(a[i]);  **end**;  i:=2;  **if** a[1]<a[2] **then**  writeln('1')  **else begin**  **while** (i<=N-1) **and** (**not**((a[i-1]>a[i]) **and** (a[i]<a[i+1]))) **do**  inc(i);  Writeln(i);  **end**;  **end**. | #include <iostream>  using namespace std;  int main(){  int \*x,i, n;  cin >> n;  x = new int[n];  for (i = 0; i <n; ++i) {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  if (x[0] < x[1])  cout << "1" << endl;  else {  i = 1;  while ((i<n-1) && !((x[i-1]> x[i])  && (x[i] < x[i + 1])))  ++i;  cout << i + 1 << endl;  }  system("pause");  return 0;  } |

1. Дан массив размера *N*. Найти количество участков, на которых его элементы монотонно возрастают.

| PascalABC.NET |
| --- |
| **Var**  a: **array of** integer;  N, i, Num: Integer;  **begin**  a[1] := 0;  Write('N: ');  Readln(N);  a:=**new** integer[N + 1];  **for** i := 1 **to** N **do**  **begin**  readln(a[i]);  **end**;  Num := 0;  **for** i := 3 **to** N **do**  **if** (a[i - 2] < a[i - 1]) **and not** (a[i - 1] < a[i]) **then**  inc(Num);  **if** a[N - 1] < a[N] **then**  inc(Num);  Writeln(Num);  **End**. |
| C++ |
| #include <iostream>  using namespace std;  int main(){  int \*x,i, n, num=0;  cin >> n;  x = new int[n];  for (i = 0; i <n; ++i)  {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  for (i = 2; i<n; ++i)  if ((x[i - 2]<x[i - 1]) && !(x[i - 1]<x[i]))  ++num;  if (x[n - 2]<x[n - 1])  ++num;  cout << num << endl;  system("pause");  return 0;  } |

1. Дан целочисленный массив размера *N*, все элементы которого упорядочены (по возрастанию или по убыванию). Найти количество различных элементов в данном массиве.

| PascalABC.NET |
| --- |
| **Var**  i, N, Q: integer;  A: **array of** integer;  **begin**  write('N= ');  readln(N);  a:=**new** integer[N + 1];  Q := 1;  **for** i := 1 **to** N **do**  **begin**  read(a[i]);  **if** (i > 1) **and** (a[i] <> a[i - 1]) **then** inc(Q)  **end**;  writeln;  writeln('Различных чисел: ', Q)  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main(){  int \*x, i, n, k = 0;  bool f;  cin >> n;  x = new int[n];  for (i = 0; i <n; ++i) {  x[i] = rand() % 5;  cout << x[i] << " ";  }  cout << endl;  for ( i = 0; i < n; i++){  int j = 0;  while (j < i && x[j] != x[i])  ++j;  k += j == i;  }  cout << k << endl;  delete x;  system("pause");  return 0;  } |

1. Дано число *R* и массив размера *N*. Найти два различных элемента массива, сумма которых наиболее близка к числу *R*, и вывести эти элементы в порядке возрастания их индексов.

| PascalABC.NET |
| --- |
| **Var**  A: **array of** real;  R, MinDiffSum, c: real;  i, j, N, i1, i2: byte;  **begin**  write('N = ');  readln(N);  a := **new** real[N + 1];  write('R = ');  readln(R);  **for** i := 1 **to** N **do**  **begin**  a[i] := random(10)+1;  write(' ', a[i]:0:2);  **if** i = 2 **then**  **begin**  MinDiffSum := abs(a[2] + a[1] - R);  i1 := 1;  i2 := 2;  **end**  **else**  **for** j:=1 **to** i-1 **do**  **begin**  c := abs(a[j] + a[i] - R);  **if** c < MinDiffSum **then**  **begin**  MinDiffSum := c;  i1 := j;  i2 := I;  **end**  **end**  **end**;  Writeln;  writeln('a[1]:=',a[i1]:0:2, ' ', 'a[2]:=',a[i2]:0:2);  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main(){  int \*x, i,j, r,n, k = 0, k2=1;  bool f;  cin >> n>>r;  x = new int[n];  for (i = 0; i <n; ++i) {  x[i] = rand() % 5;  cout << x[i] << " ";  }  cout << endl;  for (i = 0; i<n - 1; ++i) {  for (j= i + 1; j<n; ++j) {  if (abs(x[i] + x[j] - r)<abs(x[k] + x[k2] - r)) {  k = i;  k2 = j;  }  }  }  cout<<k + 1<<" "<<k2 + 1;  delete x;  system("pause");  return 0;  } |

1. Дан целочисленный массив размера *N*. Если он является *перестановкой*, то есть содержит все числа от 1 до *N*, то вывести 0; в противном случае вывести номер первого недопустимого элемента.

| PascalABC.NET |
| --- |
| **Var**  a: **array of** integer;  N, i, i2: Integer;  check: Boolean;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  **for** i := 1 **to** N **do**  **begin**  check := False;  **for** i2 := 1 **to** N **do**  **if** i = a[i2] **then**  check := True;  **if not** (check) **then**  **begin**  Writeln(i);  **break**;  **end**;  **end**;  **if** check **then**  Writeln(0);  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main() {  int \*x, i, j, n;  cin >> n;  x = new int[n];  for (i = 0; i < n; ++i) {  x[i] = rand() % 10;  cout << x[i] << " ";  }  cout << endl;  for (i = 1; i <= n; ++i) {  for (j = 0; j < n; ++j) {  if (x[j] == i) {  j = n + 100;  break;  }  }  if (j != n + 100) {  cout << i << endl;  break;  }  }  if (i == n + 1)  cout << 0 << endl;  system("pause");  return 0;  } |

## **Работа с несколькими массивами**

1. Даны два массива *A* и *B* одинакового размера *N*. Сформировать новый массив *C* того же размера, каждый элемент которого равен максимальному из элементов массивов *A* и *B* с тем же индексом.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a, b, c: **array of** real;  N, k: Integer;  **begin**  Write('N: ');  Readln(N);  a:=**new** integer[N+1];  b:=**new** integer[N+1];  c:=**new** integer[N+1];  Writeln('A: ');  **for** k := 1 **to** N **do**  **begin**  write(k, ' :');  readln(a[k]);  **end**;  Writeln('B: ');  **for** k := 1 **to** N **do**  **begin**  write(k, ' :');  readln(b[k]);  **end**;  **for** k := 1 **to** N **do**  c[k] := max(a[k], b[k]);  Writeln('C: ');  **for** k := 1 **to** N **do**  writeln(k, ' :', c[k]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*x, \*y, \*z, i, j, n;  cin >> n;  x = new int[n];  y = new int[n];  z = new int[n];  for (i = 0; i < n; ++i) {  x[i] = rand() % 50;  y[i] = rand() % 60;  cout << x[i] << " ";  }  cout << endl;  for (i = 0; i < n; ++i)  cout << y[i] << " ";  for (int k = 0; k<n; ++k){  if (x[k]>y[k])  z[k] = x[k];  else z[k] = y[k];  }  cout << endl;  for (i = 0; i < n; ++i)  cout << z[i] << " ";  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func MaxInt32(a,b int) int{  if a>b{  return a  }else{  return b  }  }  func main() {  var n int  fmt.Print("n->")  fmt.Scan(&n)  var x=make([]int, n)  var y=make([]int, n)  var z=make([]int, n)  for i := 0; i < n; i++ {  x[i]=rand.Intn(50);  y[i]=rand.Intn(60);  z[i]=MaxInt32(x[i],y[i]);  }  fmt.Println(x)  fmt.Println(y)  fmt.Println(z)  } | |

## **Преобразование массива**

При выполнении заданий из данного пункта не следует использовать вспомогательные массивы.

### **Изменение элементов массива**

1. Дан массив *A* размера *N* и целое число *K* (1 ≤ *K* ≤ *N*). Преобразовать массив, увеличив каждый его элемент на исходное значение элемента *AK*.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** integer;  N, K, i, Ak: Integer;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  Write('K: ');  Readln(K);  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  Ak := a[k];  **for** i := 1 **to** N **do**  a[i] := a[i] + Ak;  **for** i := 1 **to** N **do**  writeln(i, ' :', a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*x, i, k, n;  cin >> n >> k;  x = new int[n];  for (i = 0; i < n; ++i) {  x[i] = rand() % 50;  cout << x[i] << " ";  }  cout << endl;  for (i = 0; i < n; ++i) {  x[i] += x[k - 1];;  }  for (i = 0; i < n; ++i)  cout << x[i] << " ";  system("pause");  return 0;  } |

1. Дан массив размера *N* (*N* — четное число). Поменять местами первую и вторую половины массива.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a : **array of** integer;  N,i:Integer;    **begin**  Write('N: ');  Readln(N);  a:=**new** integer[N+1];  Writeln('A: ');  **for** i:=1 **to** N **do**  **begin**  readln(a[i]);  **end**;  i:=1;  **while** i<=N/2 **do**  **begin**  swap(a[i],a[(N **div** 2)+i]);  i:=i+1;  **end**;  **for** i:=1 **to** N **do**  writeln(i,' :',a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 50;  cout << a[i] << " ";  }  cout << endl;  for (i = 0; i<n / 2; ++i) {  a[i] += a[n / 2 + i];  a[n/2+i]=a[i]-a[n/2+i];  a[i]=a[i]-a[n / 2 + i];  }  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан массив размера *N*. Осуществить *сдвиг* элементов массива вправо на одну позицию (при этом *A*1 перейдет в *A*2, *A*2 — в *A*3, …, *AN*–1 — в *AN*,   
   a исходное значение последнего элемента будет потеряно). Первый элемент полученного массива положить равным 0.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  a: **array of** Integer;  N, i: Integer;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  **for** i := N **downto** 2 **do**  a[i] := a[i - 1];  A[1] := 0;  **for** i := 1 **to** N **do**  writeln(i, ':', a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 50;  cout << a[i] << " ";  }  cout << endl;  for (i = n - 1; i >= 1; --i)  a[i] = a[i - 1];  a[0] = 0;  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан массив размера *N*, все элементы которого, кроме последнего, упорядочены по возрастанию. Сделать массив упорядоченным, переместив последний элемент на новую позицию.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** Integer;  N, i: Integer;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  **for** i:=N-1 **downto** 1 **do**  **if** a[i] > a[i + 1] **then**  swap(a[i], a[i + 1]);  **for** i := 1 **to** N **do**  writeln(i, ' :', a[i]);  **end**. | #include <iostream>  using namespace std;  void swap(float &a, float &b) {  int h = a;  a = b;  b = h;  }  int main() {  int \*a, i, n;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 50;  cout << a[i] << " ";  }  cout << endl;  for (i = n - 2; i >= 0; --i) {  if (a[i] > a[i + 1]) {  swap(a[i], a[i + 1]);  }  else  break;  }  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

### **Удаление и вставка элементов**

1. Дан массив размера *N* и целое число *K* (1 ≤ *K* ≤ *N*). Удалить из массива элемент с порядковым номером *K*.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** Integer;  N, K, i: Integer;  **begin**  Write('N: ');  Readln(N);  a:=**new** integer[N + 1];  Write('K: ');  Readln(K);  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  N := N - 1;  **for** i := K **to** N **do**  a[i] := a[i + 1];  **for** i := 1 **to** N **do**  writeln(i,' :',a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, k;  cin >> n>>k;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 50;  cout << a[i] << " ";  }  cout << endl;  --n;  for (i = k - 1; i<n; ++i) {  a[i] = a[i + 1];  }  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан целочисленный массив размера *N*. Удалить из массива все нечетные числа и вывести размер полученного массива и его содержимое.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** Integer;  N, i, k: Integer;  **begin**  Readln(N);  a := **new** integer[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  k := 0;  **for** i := 1 **to** n **do**  **if not** odd(a[i]) **then**  **begin**  inc(k);  a[k] := a[i];  **end**;  Writeln(N);  **for** i := 1 **to** k **do**  writeln(i, ' :', a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, k=0;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 50;  cout << a[i] << " ";  }  cout << endl;  for (i = 0; i<n; ++i) {  if (a[i] % 2 == 0) {  a[k] = a[i];  ++k;  }  }  for (i = 0; i < k; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан целочисленный массив размера *N*. Удалить из массива все элементы, встречающиеся менее трех раз, и вывести размер полученного массива и его содержимое.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** Integer;  N, i, i2, k, num: Integer;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  **repeat**  k := 11;  **for** i := 1 **to** N **do**  **begin**  Num := 0;  **for** i2 := 1 **to** N **do**  **if** a[i]=a[i2] **then**  inc(num);  **if** num<3 **then** k := i;  **end**;  **if** k <> 11 **then**  **begin**  N := N - 1;  **for** i := K **to** N **do**  a[i] := a[i + 1];  **end**;  **until** (k = 11);  Writeln(N);  **for** i := 1 **to** N **do**  writeln(i, ' :', a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, k = -1, num, j;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 10;  cout << a[i] << " ";  }  cout << endl;  for (i = 0; i < n; ++i) {  for (j = 0; j <= k; ++j) {  if (a[i] == a[j])  break;  }  if (j != k + 1) {  ++k;  a[k] = a[i];  }  else {  num = 0;  for (j = i; j < n; ++j) {  if (a[j] == a[i])  ++num;  }  if (num >= 3) {  ++k;  a[k] = a[i];  }  }  }  ++k;  for (i = 0; i < k; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан массив размера *N* и целое число *K* (1 ≤ *K* ≤ *N*). Перед элементом массива с порядковым номером *K* вставить новый элемент с нулевым значением.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** Integer;  N, i, K: Integer;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  Write('K: ');  Readln(K);  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  N := N + 1;  **for** i := N **downto** K + 1 **do**  a[i] := a[i - 1];  a[k] := 0;  Writeln(N);  **for** i := 1 **to** N **do**  writeln(i, ' :', a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, k;  cin >> n >> k;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 10;  cout << a[i] << " ";  }  cout << endl;  n++;  for (i = n - 1; i >= k; --i)  a[i] = a[i - 1];  a[k - 1] = 0;  for (i = 0; i <n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан массив размера *N*. Перед каждым положительным элементом массива вставить элемент с нулевым значением.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  a, b: **array of** Integer;  N, Nk, i, k: Integer;  **begin**  Write('N: ');  Readln(N);  a := **new** integer[N + 1];  b := **new** integer[2\*N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  Nk := N;  k := 0;  **for** i := 1 **to** N **do**  **begin**  **if** a[i] > 0 **then**  **begin**  Nk := Nk + 1;  b[k + 1] := a[i];  b[k] := 0;  K := k + 1;  **end**  **else**  b[k] := a[i];  K := k + 1;  **end**;  N := Nk;  a := b;  Writeln(N);  **for** i := 1 **to** N **do**  writeln(i, ' :', a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, j;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 10-3;  cout << a[i] << " ";  }  cout << endl;  for (i = n - 1; i >= 0; --i) {  if (a[i] > 0) {  ++n;  for (j = n - 1; j > i; --j)  a[j] = a[j - 1];  a[i] = 0;  }  }  for (i = 0; i <n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

### **Сортировка массива**

1. Дан массив *A* размера *N* (≤ 6). Упорядочить его по возрастанию методом сортировки *простым обменом* (*«пузырьковой» сортировкой*): просматривать массив, сравнивая его соседние элементы (*A*1 и *A*2, *A*2 и *A*3   
   и т. д.) и меняя их местами, если левый элемент пары больше правого; повторить описанные действия *N* – 1 раз. Для контроля за выполняемыми действиями выводить содержимое массива после каждого просмотра. Учесть, что при каждом просмотре количество анализируемых пар можно уменьшить на 1.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array of** real;  N, N2, i, i2, i3: integer;  **begin**  Write('N: ');  Readln(N);  a := **new** real[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  N2 := N;  **for** i := 1 **to** N - 1 **do**  **begin**  i2 := 1;  N2 := N2 - 1;  **while** (i2 <= N2) **do**  **begin**  **if** a[i2]>a[i2+1] **then**  swap(a[i2],a[i2+1]);  inc(i2);  **end**;  **for** i3 := 1 **to** N **do**  writeln(a[i3]:7:0);  **end**;  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, j;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 10 - 3;  cout << a[i] << " ";  }  cout << endl;  for (int i = 1; i < n; ++i)  {  for (int j = 0; j < n - i; j++)  {  if (a[j] < a[j + 1])  {  int temp = a[j];  a[j] = a[j + 1];  a[j + 1] = temp;  }  }  }  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан массив *A* размера *N* (≤ 6). Упорядочить его по возрастанию методом сортировки *простыми вставками*: сравнить элементы *A*1 и *A*2 и, при необходимости меняя их местами, добиться того, чтобы они оказались упорядоченными по возрастанию; затем обратиться к элементу *A*3 и переместить его в левую (уже упорядоченную) часть массива, сохранив ее упорядоченность; повторить этот процесс для остальных элементов, выводя содержимое массива после обработки каждого элемента (от 2-го до *N*-го). При выполнении описанных действий удобно использовать прием «барьера», записывая очередной элемент перед его обработкой в дополнительный элемент массива *A*0.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  a: **array of** real;  N, i, i2, i3: integer;  **begin**  Write('N: ');  Readln(N);  a := **new** real[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  **if** a[1] > a[2] **then**  swap(a[1], a[2]);  **for** i := 3 **to** N **do**  **begin**  a[0] := a[i];  **for** i2:=i-1 **downto** 0 **do**  **if** a[0] < a[i2] **then**  a[i2 + 1] := a[i2]  **else**  **begin**  a[i2 + 1] := a[0];  **break**;  **end**;  **for** i3 := 1 **to** N **do**  writeln(i3,':',a[i3]);  writeln;  **end**;  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, j, temp;  cin >> n;  a = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 10 - 3;  cout << a[i] << " ";  }  cout << endl;  if (a[0] > a[1]) {  temp = a[0];  a[0] = a[1];  a[1] = temp;  }  for (i = 2; i <= n - 1; ++i) {  temp = a[i];  for (j = i - 1; j >= 0; --j)  if (temp<a[j]) a[j+1]=a[j];  else  break;  a[j + 1] = temp;  }  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

## **Серии целых чисел**

1. Дан целочисленный массив *A* размера *N*. Назовем *серией* группу подряд идущих одинаковых элементов, а *длиной серии* — количество этих элементов (длина серии может быть равна 1). Сформировать два новых целочисленных массива *B* и *C* одинакового размера, записав в массив *B* длины всех серий исходного массива, а в массив *C* — значения элементов, образующих эти серии.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a, b, c: **array of** real;  N, i, k: integer;  **begin**  Write('N: ');  Readln(N);  a:=**new** real[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  k := 1;  b[k] := 1;  c[k] := a[k];  **for** i := 2 **to** N **do**  **begin**  **if** a[i-1]=a[i] **then**  b[k] := b[k] + 1  **else**  **begin**  inc(k);  b[k] := 1;  c[k] := a[i];  **end**;  **end**;  **for** i := 1 **to** k **do**  writeln(c[i],' - ',b[i] );  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a,\*b,\*c, i, n, j, k=0;  cin >> n;  a = new int[n];  b = new int[n];  c = new int[n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 5;  cout << a[i] << " ";  }  cout << endl;  b[k] = 1;  c[k] = a[k];  for (i = 1; i < n; ++i) {  if (a[i - 1] == a[i]) ++b[k];  else {  ++k;  b[k] = 1;  c[k] = a[i];  }  }  for (i = 0; i < k; ++i)  cout <<b[i] << " ";  cout << endl;  for (i = 0; i < k; ++i)  cout << c[i] << " ";  system("pause");  return 0;  } |

1. Дан целочисленный массив размера *N*. Вставить перед каждой его серией элемент с нулевым значением.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a, b: **array of** real;  N, i, k: integer;  **begin**  Write('N: ');  Readln(N);  a:=**new** real[N + 1];  b:=**new** real[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  b[1] := 0;  k := 2;  b[k] := a[1];  **for** i := 2 **to** N **do**  **begin**  inc(k);  **if** a[i-1] <> a[i] **then**  **begin**  b[k] := 0;  inc(k);  **end**;  b[k] := a[i];  **end**;  a := b;  **for** i := 1 **to** k **do**  writeln(a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a,\*b, i, n, k=1;  cin >> n;  a = new int[n];  b = new int[n\*2];  for (i = 0; i < n; ++i) {  a[i] = rand() % 5;  cout << a[i] << " ";  }  cout << endl;  b[0] = 0;  b[k] = a[1];  for (i = 1; i<n; ++i) {  ++k;  if (a[i - 1] != a[i]) {  b[k] = 0;  ++k;  }  b[k] = a[i];  }  for (i = 0; i < k; ++i)  cout <<b[i] << " ";  system("pause");  return 0;  } |

1. Дано целое число *K* (> 0) и целочисленный массив размера *N*. Преобразовать массив, удвоив длину его серии с номером *K*. Если серий в массиве меньше *K*, то вывести массив без изменений.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a, b: **array of** real;  N, K, i, i2, ki: integer;  **begin**  Write('N: ');  Readln(N);  a := **new** real[N + 1];  b := **new** real[N + 1];  Write('K: ');  Readln(K);  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  i2 := 0;  ki := 1;  **for** i := 1 **to** N - 1 **do**  **begin**  **if** ki = K **then**  **begin**  inc(i2);  b[i2] := a[i];  **end**;  inc(i2);  b[i2] := a[i];  **if** a[i] <> a[i + 1] **then**  inc(ki);  **end**;  **if** ki = K **then**  **begin**  inc(i2);  b[i2] := a[i];  **end**;  inc(i2);  b[i2] := a[i];  a := b;  **for** i := 1 **to** i2 **do**  writeln(a[i]);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, k = 1;  cin >> n;  a = new int[2 \* n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 5;  cout << a[i] << " ";  }  cout << endl;  int num = 1;  for (i=0;(i<n-1)&&(num<k);++i) {  if (a[i] != a[i + 1])  ++num;  }  int begine = i;  for (i=begine; (i<n-1); ++i) {  if (a[i] != a[i + 1]) break;  }  int ende = i;  int i2;  for (i2=1;i2<=ende-begine+1; ++i2) {  n++;  for (i = n - 1; i > ende; i--) {  a[i] = a[i - 1];  }  }  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

1. Дан целочисленный массив размера *N*. Преобразовать массив, увеличив его последнюю серию наибольшей длины на один элемент.

| PascalABC.NET | C++ |
| --- | --- |
| **Const**  col = 20;  **var**  a, c, temp: **array** [1..col] **of**  real;  x, y: **array** [1..col] **of**  integer;  N, i, i2, i3, ns, ms:  integer;  **begin**  Write('N: ');  Readln(N);  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, ' :');  readln(a[i]);  **end**;  ns := 1;  x[ns] := 1;  y[ns] := ns;  ms := ns;  c[ns] := a[1];  **for** i := 2 **to** N **do**  **begin**  **if** a[i - 1] = a[i] **then**  **begin**  x[ns] := x[ns] + 1  **end**  **else begin**  **if** x[ns]>=x[ms] **then**  ms:=ns;  inc(ns);  x[ns] := 1;  y[ns] := i;  c[ns] := a[i];  **end**;  **end**;  **if** x[ns] >= x[ms] **then**  ms := ns;  y[ns + 1] := N + 1;  x[ms] := x[ms] + 1;  i3 := 0;  **for** i := 1 **to** ns **do**  **for** i2 := 1 **to** x[i] **do**  **begin**  inc(i3);  temp[i3] := c[i];  **end**;  a := temp;  **for** i := 1 **to** i3 **do**  writeln(a[i] );  **end**. | #include <iostream>  using namespace std;  int main() {  int \*a, i, n, k = 1;  cin >> n;  a = new int[2 \* n];  for (i = 0; i < n; ++i) {  a[i] = rand() % 5;  cout << a[i] << " ";  }  cout << endl;  int len = 1, maxlen = 1,  endmaxseries = 1;  for (i = 1; i < n; ++i) {  if (a[i - 1] != a[i]) {  if (len >= maxlen) {  maxlen = len;  endmaxseries=i-1;  }  len = 0;  }  ++len;  }  if (len > maxlen) {  maxlen = len;  endmaxseries = i - 1;  }  for (i=++n-1;i>endmaxseries;--i)  a[i] = a[i - 1];  for (i = 0; i < n; ++i)  cout << a[i] << " ";  system("pause");  return 0;  } |

## **Множества точек на плоскости**

Для хранения данных о каждом наборе точек следует использовать по два массива: первый массив для хранения абсцисс, второй — для хранения ординат. Можно также использовать массив *записей* с двумя полями.

1. Дано множество *A* из *N* точек на плоскости и точка *B* (точки заданы своими координатами *x*, *y*). Найти точку из множества *A*, наиболее близкую к точке *B*. *Расстояние* *R* между точками с координатами (*x*1, *y*1) и (*x*2, *y*2) вычисляется по формуле:

| PascalABC.NET |
| --- |
| **Var**  Ax, Ay: **array of** real;  Bx, By, R, Rmin: real;  N, i, Amin: integer;  **begin**  Write('N: ');  Readln(N);  Ax:=**new** real[N + 1];  Ay:=**new** real[N + 1];  Writeln('A: ');  **for** i := 1 **to** N **do**  **begin**  write(i, '.x :');  readln(Ax[i]);  write(i, '.y :');  readln(Ay[i]);  **end**;    Writeln('B');  Write('x: ');  Readln(Bx);  Write('y: ');  Readln(By);    Rmin := sqrt(sqr(AX[1] - Bx) + sqr(Ay[1] - By));  Amin := 1;    **for** i := 2 **to** N **do**  **begin**  R := sqrt(sqr(Ax[i] - Bx) + sqr(Ay[i] - By));  **if** R < Rmin **then**  **begin**  Rmin := R;  Amin := i;  **end**;  **end**;  Writeln(Amin);  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main() {  float a[100][2];  int n, amin = INT32\_MAX;  float bx, by;  cin >> n >> bx >> by;  int i;  for (i = 0; i < n; ++i) {  cout << (i + 1) << endl;  cout << "x=";  cin >> a[i][0];  cout << "y=";  cin >> a[i][1];  }  float r, rmin=32875;  for (i = 0; i < n; ++i) {  r=sqrt(pow(bx-a[i][0],2)+pow(by- a[i][1], 2));  if (r < rmin || i == 0) {  rmin = r;  amin = i;  }  }  cout<<amin+1<<" "<< a[amin][0]  << " " << a[amin][1];  system("pause");  return 0;  } |

1. Даны множества *A* и *B*, состоящие соответственно из *N*1 и *N*2 точек (точки заданы своими координатами *x*, *y*). Найти минимальное расстояние между точками этих множеств и сами точки, расположенные на этом расстоянии (вначале выводится точка из множества *A*, затем точка из множества *B*).

| PascalABC.NET |
| --- |
| **Var**  Ax,Ay: **array of** real;  R,Rmax:real;  N,i,i2,Point1,Point2: integer;    **begin**  Write('N: ');  Readln(N);  Ax:=**new** real[N+1];  Ay:=**new** real[N+1];  Writeln('A: ');  **for** i:=1 **to** N **do**  **begin**  write(i,'.x :');  readln(ax[i]);  write(i,'.y :');  readln(Ay[i]);  **end**;  Rmax:=0;  Point1:=0;  Point2:=0;  **for** i:=1 **to** N-1 **do**  **for** i2:=i+1 **to** N **do**  **begin**  R:=sqrt(sqr(Ax[i]-Ay[i2])+sqr(Ax[i]-Ay[i2]));  writeln(R);  **if** (R > Rmax) **then**  **begin**  Rmax:=R;  Point1:=i;  Point2:=i2;  **end**;  **end**;  Writeln ('(',Ax[Point1],':',Ay[Point1],')(',Ax[Point2],':',  Ay[Point2],')');  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main() {  float a[100][2];  int i, n;  cin >> n;  for (i = 0; i < n; ++i)  cin >> a[i][0] >> a[i][1];  int a1 = 0, a2 = 0, i2;  float r, rmax = 0;  for (i = 0; i < n; ++i) {  for (i2 = i + 1; i2 < n; ++i2) {  r=sqrt(pow(a[i][0]-a[i2][0],2)+pow(a[i][1]-a[i2][1],2));  if (r > rmax) {  rmax = r;  a1 = i;  a2 = i2;  }  }  }  cout << a1 + 1 << " " << a[a1][0] << " " << a[a1][1];  cout << a2 + 1 << " " << a[a2][0] << " " << a[a2][1];  system("pause");  return 0;  } |

# **Двумерные массивы (матрицы): группа Matrix**

Условие вида «дана матрица размера M×N» означает, что вначале дается *фактический размер* двумерного массива-матрицы (количество строк *M* и количество столбцов *N*), а затем приводятся элементы этого массива (количество элементов равно *M*·*N*). Если в задании явно не указывается, какие значения могут принимать размеры исходной матрицы, то предполагается, что и число строк, и число столбцов может меняться в пределах от 2 до 10. Порядковые номера начальной строки и начального столбца матрицы считаются равными 1. Ввод и вывод элементов матрицы осуществляются *по строкам*.

*Квадратной матрицей порядка* *M* называется двумерный массив-матрица размера

Если в задании, связанном с созданием (преобразованием) матрицы, не описан результирующий набор данных, то предполагается, что этим набором является созданная (преобразованная) матрица, и необходимо вывести все ее элементы.

## **Формирование матрицы и вывод ее элементов**

В заданиях на формирование матрицы предполагается, что размер результирующей матрицы не превосходит 10.

1. Даны целые положительные числа *M* и *N*. Сформировать целочисленную матрицу размера , у которой все элементы *I*-й строки имеют значение 10·*I* (*I* = 1, …, *M*).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  i, j: Integer;  **begin**  **var** m := ReadInteger('m');  **var** n:= ReadInteger('n: ');  **var** a: **array** [,] **of** integer  :=**new** integer[m + 1, n + 1];  **for** i := 1 **to** M **do**  **for** j := 1 **to** N **do**  a[i, j] := i \* 10;  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  Write(a[i, j], '');  writeln();  **end**;  **end**. | | #include <iostream>  using namespace std;  int main() {  int \*\*x, n, m, i, j;  cin >> m >> n;  x = new int\*[m];  for (i = 0; i < m; i++)  x[i] = new int[n];  for (i = 0; i < m; i++)  for (j = 0; j < n; j++)  x[i][j] = 10 \* (i+1);  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout<<x[i][j] << " ";  cout << endl;  }  system("pause");  return 0;  } | |
| Python | C# | | |
| from random import random  a=[]  n=int(input(""))  for i in range(1,n+1):  x=[]  for j in range (1,n+1):  x.append(i\*10))  a.append(x);  for row in a:  print(row) | static void Main(string[] args)  {  int m, n, i, j;  m=Convert.ToInt32(Console.ReadLine());  n = Convert.ToInt32(Console.ReadLine());  int [,]x=new Int32[m,n];  for (i=0; i<m; i++)  for (j=0; j<n; j++)  x[i, j]=(i+1)\*10;  for (i=0; i<m; i++) {  for (j=0; j<n; j++)  Console.Write(x[i, j]+" ");  Console.WriteLine();  }  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args){  Scanner sc=new Scanner(System.in);  int m, n, i, j;  m=sc.nextInt();  n=sc.nextInt();  int [][] x=new int[m][n];  Random rnd=new Random();  for (i=0; i<m; i++)  for (j=0; j<n; j++)  x[i][j]=rnd.nextInt(50);  for (i=0; i<m; i++){  for (j=0; j<n; j++)  System.out.print(x[i][j]+" ");  System.out.println();  }  } | | | package main  import "fmt"  func main() {  var n,m int  x := [10][10]int{}  fmt.Print("n, m(<10)")  fmt.Scan(&n, &m)  for i:=0; i<n; i++ {  for j:=0; j<m; j++ {  x[i][j] = 10\*(i+1)  }  }  for i:=0; i<n; i++ {  for j:=0; j<m; j++ {  fmt.Printf("%v ",x[i][j])  }  fmt.Println()  }  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** m = reader.nextInt()  **var** n = reader.nextInt()  **val** a = Array(m) **{** *arrayOfNulls*<Int>(n) **}  for** (i **in** (0..m - 1)) {  **for** (j **in** (0..n - 1))  a[i][j] = 10 \* (i + 1)  }  **for** (i **in** (0..m - 1)) {  **for** (j **in** (0..n - 1))  *print*(a[i][j].*toString*() + **" "**)  *println*()  } } | | | |
| VB.NET | | | |
| Sub Main()  Dim a(,) As Integer  Dim i, j, m, n As Integer  m = Convert.ToInt32(Console.ReadLine())  n = Convert.ToInt32(Console.ReadLine())  a = New Integer(m, n) {}  For i = 0 To m - 1  For j = 0 To n - 1  a(i, j) = 10 \* (i + 1)  Next  Next  For i = 0 To m - 1  For j = 0 To n - 1  Console.Write("{0} ", a(i, j))  Next  Console.WriteLine()  Next  Console.ReadKey()  End Sub | | | |

1. Даны целые положительные числа *M*, *N* и набор из *M* чисел. Сформировать матрицу размера M×N, у которой в каждом столбце содержатся все числа из исходного набора (в том же порядке).

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array** [1..10, 1..10] **of** integer;  M, N, i, j: Integer;  **begin**  Write('N,M: ');  Readln(N, M);    Writeln('Nabor:');  **for** i := 1 **to** M **do**  **begin**  Write(i, ': ');  readln(a[i, 1]);  **end**;    **for** i := 1 **to** M **do**  **for** j := 2 **to** N **do**  a[i, j] := a[i, j - 1];    **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  Write(a[i, j], ' ');  writeln;  **end**;  **end**. | #include <iostream>  using namespace std;  int main() {  int \*\*x, n, m, i, j;  cin >> m >> n;  x = new int\*[m];  for (i = 0; i < m; i++)  x[i] = new int[n];  for (i = 0; i < m; i++)  cin >> x[i][0];  for (i = 0; i < m; i++)  for (j = 1; j < n; j++)  x[i][j] = x[i][j - 1];  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import "fmt"  func main() {  var n,m int  x := [10][10]int{}  fmt.Print("m, n(<10)")  fmt.Scan(&m, &n)  for j:=0; j<n; j++ {  fmt.Scan(&x[0][j])  }  for i:=1; i<m; i++ {  for j:=0; j<n; j++ {  x[i][j] = x[i-1][j]  }  }  for i:=0; i<m; i++ {  for j:=0; j<n; j++ {  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  } | |

1. Дана квадратная матрица *A* порядка *M*. Начиная с элемента *A*1,1, вывести ее элементы следующим образом («*уголками*»): все элементы первой строки; элементы последнего столбца, кроме первого (уже выведенного) элемента; оставшиеся элементы второй строки; оставшиеся элементы предпоследнего столбца и т. д.; последним выводится элемент *AM*,1.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array** [0..10, 0..10] **of** integer;  M, i, j, i2: Integer;  **begin**    Write('M: ');  Readln(M);  M := M - 1;  **for** i := 0 **to** M **do**  **begin**  **for** j := 0 **to** M **do**  **begin**  a[i, j] := Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;  **for** i := 0 **to** M **do**  **begin**  **for** j := 0 **to** M - i **do**  write(' ', a[i, j]);  **for var** k := i + 1 **to** M **do**  write(' ', a[k, M - i]);  writeln('');  **end**;  **end**. | #include <iostream>  using namespace std;  int main() {  int \*\*x, m, i, j;  cin >> m;  m = m - 1;  x = new int\*[m+2];  for (i = 0; i <=m; i++)  x[i] = new int[m+2];  for (i = 0; i <=m; i++)  for (j = 0; j <= m; j++)  cin>>x[i][j];  for (i = 0; i <= m; i++) {  for (j = 0; j <=m; j++)  cout << x[i][j] << " ";  cout << endl;  }  cout << endl;  for (i = 0; i <= m; i++) {  for (j = 0; j <=m-i; j++)  cout << " "<<x[i][j];  for (int k=i+1; k<=m; k++)  cout << " " << x[k][m-i];  cout << endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,i,j,k int  x := [10][10]int{}  fmt.Print("m(<10)")  fmt.Scan(&m)  for i=0; i<m; i++ {  for j=0; j<m; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  for i=0; i<m; i++ {  for j=0; j<m-i; j++ {  fmt.Printf("%v ", x[i][j])  }  for k=i+1; k<m; k++ {  fmt.Printf("%v ", x[k][m-1-i])  }  fmt.Println()  }  } | |

1. Дана квадратная матрица *A* порядка *M* (*M* — нечетное число). Начиная с элемента *A*1,1 и перемещаясь по часовой стрелке, вывести все ее элементы *по спирали*: первая строка, последний столбец, последняя строка в обратном порядке, первый столбец в обратном порядке, оставшиеся элементы второй строки и т. д.; последним выводится центральный элемент матрицы.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array** [0..10, 0..10]  **of** integer;  M, i, j, k: Integer;  **begin**  Write('M: ');  Readln(M);  M := M - 1;  **for** i := 0 **to** M **do**  **begin**  **for** j := 0 **to** M **do**  **begin**  a[i,j]:=Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;  **for** i := 0 **to** M **div** 2 **do**  **begin**  **for** j:=i **to** M-i **do**  Write(a[i, j], ' ');  **for** k:=i+1 **to** M-i **do**  Write(a[k,M-i],' ');  **for** j:=M-1-i **downto** i **do**  Write(a[M-i,j], ' ');  **for** k:=M-1-i **downto** i+1 **do**  Write(a[k, i], ' ');  writeln('');  **end**;  **end**. | #include <iostream>  using namespace std;  int main() {  int \*\*x, m, i, j, k;  cin >> m;  m = m - 1;  x = new int\*[m+2];  for (i = 0; i <=m; i++)  x[i] = new int[m+2];  for (i = 0; i <=m; i++)  for (j = 0; j <= m; j++)  x[i][j]=rand()%50;  for (i = 0; i <= m; i++) {  for (j = 0; j <=m; j++)  cout << x[i][j] << " ";  cout << endl;  }  cout << endl;  for (i = 0; i <= m/2; i++) {  for (j = i; j <= m-i; j++)  cout <<x[i][j]<<" ";  for (k=i+1; k<=m-i; k++)  cout << x[k][m-i]<<" ";  for (j=m-1-i; j>=i; j--)  cout << x[m-i][j] << " ";  for (k=m-1-i; k>=i+1; k--)  cout << x[k][i] << " ";  cout << endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,i,j,k int  x := [10][10]int{}  fmt.Scan(&m)  for i=0; i<m; i++ {  for j=0; j<m; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  for i=0; i<m/2; i++ {  for j=0; j<m-i; j++ {  fmt.Printf("%v ", x[i][j])  }  for k=i+1; k<m-i; k++ {  fmt.Printf("%v ", x[k][m-1-i])  }  for j=m-2-i;j >=i; j--{  fmt.Printf("%v ", x[m-1-i][j])  }  for k=m-2-i; k>=i+1; k-- {  fmt.Printf("%v ", x[k][i])  }  fmt.Println()  }  } | |

## **Анализ элементов матрицы**

1. Дана матрица размера M×N и целое число *K* (1 ≤ *K* ≤ *M*). Найти сумму и произведение элементов *K*-й строки данной матрицы.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array** [1..10, 1..10]  **of** integer;  M, N, K, i, j, i2: Integer;  **begin**  Write('M,N,K: ');  Readln(M, N, K);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i,j]:=Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;    **var** S := 0;  **var** P := 1;  **for** j := 1 **to** N **do**  **begin**  S := S + a[K, j];  P := P \* a[K, j];  **end**;  Write('S=', S, 'P=', P);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*\*x, m, n,k, i, j, S=0, P=1;  cin >> m>>n>>k;  x = new int\*[m];  for (i = 0; i <m; i++)  x[i] = new int[n];  for (i = 0; i <m; i++)  for (j = 0; j <n; j++)  x[i][j]=rand()%50;  for (i = 0; i <m; i++) {  for (j = 0; j <n; j++)  cout << x[i][j] << " ";  cout << endl;  }  cout << endl;  for (j = 0; j < n; j++) {  S = S + x[k-1][j];  P = P\*x[k-1][j];  }  cout << S << " " << P << endl;  system("pause");  return 0;  } |
| C# | |
| static void Main(){  int m, n, k, i, j, sum = 0, mult = 1;  Console.Write("M=");  m = Convert.ToInt32(Console.ReadLine());  Console.Write("N=");  n = Convert.ToInt32(Console.ReadLine());  Console.Write("K=");  k = Convert.ToInt32(Console.ReadLine());  // Объявляем массив  int[, ] A = new int[m, n];  // Вводим массив  for (i = 0; i < m; i++)  {  for (j = 0; j < n; j++)  {  Console.Write("A[" + (i + 1) + "," + (j + 1) + "]=");  A[i, j] = Convert.ToInt32(Console.ReadLine());  }  }  // Считаем сумму и произведение элементов k-й строки  for (j = 0; j < n; j++)  {  sum += A[k - 1, j];  mult \*= A[k - 1, j];  }  Console.WriteLine("Сумма " + sum);  Console.WriteLine("Произведение " + mult);  }  }  } | |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,i,j,k int  x := [10][10]int{}  fmt.Print("m(<10)")  fmt.Scan(&m)  for i=0; i<m; i++ {  for j=0; j<m; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  fmt.Print("k(<10)")  fmt.Scan(&k)  var S,P int=0,1  for j=0;j<m ; j++{  S+=x[k-1][j];  P\*=x[k-1][j];  }  fmt.Printf("%d %d", S,P)  } | |

1. Дана целочисленная матрица размера M×N, элементы которой могут принимать значения от 0 до 100. Различные строки матрицы назовем *похожими*, если совпадают множества чисел, встречающихся в этих строках. Найти количество строк, похожих на первую строку данной матрицы.

| PascalABC.NET |
| --- |
| **Var**  a: **array** [1..10, 1..10] **of** integer;  Num, M, N, i, j: Integer;  L1, LM: **set of** byte;  **begin**  Write('N,M: ');  Readln(N, M);  **for** i := 1 **to** M **do**  **for** j := 1 **to** N **do**  Read(a[i, j]);  writeln;  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;  L1 := [];  **for** j := 1 **to** N **do**  L1 := L1 + [a[1, j]];  Num := 0;  **for** i := 2 **to** M **do**  **begin**  LM := [];  **for** j := 1 **to** N **do**  LM := LM + [a[i, j]];  **if** L1 = LM **then**  inc(Num);  **end**;  Writeln(Num);  **end**. |
| С++ |
| #include <iostream>  using namespace std;  void sort(int a[], int n) {  for (int i = 1; i < n; ++i)  for (int j = 0; j < n - i; j++)  if (a[j] > a[j + 1]){  int temp = a[j];  a[j] = a[j + 1];  a[j + 1] = temp;  }  }  bool isLike(int a[], int b[], int n) {  sort(a, n);  sort(b, n);  int i = 0;  while ((a[i] == b[i]) && (i<n))  i++;  return i == n;  }  int main() {  int \*\*a, m,n, i,j, k=0;  cin >> m>>n;  a = new int\*[m];  for (i = 0; i < m; i++)  a[i] = new int[n];  for (i = 0; i < m; ++i)  for (j=0; j<n; j++)  cin>>a[i][j];  cout << endl;  for (i = 0; i < m; ++i) {  for (j = 0; j < n; j++)  cout << a[i][j] << " ";  cout << endl;  }  for (i = 0; i < m; i++)  if (isLike(a[0], a[i], n))  k++;  cout <<"k="<<k<<endl;  for (i = 0; i < m; i++)  delete a[i];  delete a;  system("pause");  return 0;  } |

1. Дана целочисленная матрица размера M×N. Найти количество ее строк, все элементы которых различны.

| PascalABC.NET |
| --- |
| **Var**  a: **array** [1..10, 1..10] **of** integer;  Rez, M, N, i, j: Integer;  Row: **set of** byte;  **begin**  Write('N,M: ');  Readln(N, M);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i, j] := Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;    Rez := 0;  **for** i := 1 **to** M **do**  **begin**  j := 1;  Row := [];  **while** (**not** (a[i, j] **in** Row)) **and** (j <= N) **do**  **begin**  Row := Row + [a[i, j]];  inc(j);  **end**;  **if** J = N + 1 **then**  inc(Rez);  **end**;  Writeln(Rez);  **end**. |
| С++ |
| #include <iostream>  using namespace std;  void sort(int a[], int n) {  for (int i = 1; i < n; ++i)  for (int j = 0; j < n - i; j++)  if (a[j] > a[j + 1]){  int temp = a[j];  a[j] = a[j + 1];  a[j + 1] = temp;  }  }  bool isUnique(int a[], int n) {  sort(a, n);  int i = 1;  while ((a[i-1]<a[i]) && (i<n))  i++;  return i == n;  }  int main() {  int \*\*a, m,n, i,j, k=0;  cin >> m>>n;  a = new int\*[m];  for (i = 0; i < m; i++)  a[i] = new int[n];  for (i = 0; i < m; ++i)  for (j=0; j<n; j++)  cin>>a[i][j];  cout << endl;  for (i = 0; i < m; ++i) {  for (j = 0; j < n; j++)  cout << a[i][j] << " ";  cout << endl;  }  for (i = 0; i < m; i++)  if (isUnique(a[i], n))  k++;  cout <<"k="<<k<<endl;  for (i = 0; i < m; i++)  delete a[i];  delete a;  system("pause");  return 0;  } |

## **Преобразование матрицы**

При выполнении заданий из данного пункта не следует использовать вспомогательные двумерные массивы-матрицы.

1. Дана матрица размера M×N и целые числа *K*1 и *K*2 (1 ≤ *K*1 < *K*2 ≤ *M*). Поменять местами строки матрицы с номерами *K*1 и *K*2.

| PascalABC.NET | C++ |
| --- | --- |
| **Type**  matrix = **array** [1..10,  1..10] **of** integer;  **var**  a: matrix;  M,N,i,j,K1,K2: Integer;  **begin**  Write('N,M: ');  Readln(N, M);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i,j]:=Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;  writeln('K1,K2');  Read(K1, K2);  **for** j := 1 **to** N **do**  swap(a[K1,j], a[K2,j]);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  Write(a[i, j],' ');  **end**;  writeln();  **end**;  **end**. | #include <iostream>  using namespace std;  void swap(int &a, int &b) {  int h = a;  a = b;  b = h;  }  int main() {  int \*\*x, m, n, i, j,k1, k2;  cin >> m >> n;  x = new int\*[m];  for (i = 0; i < m; i++)  x[i] = new int[n];  for (i = 0; i < m; i++)  for (j = 0; j < n; j++)  x[i][j]=rand() % 50;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  cout << "k1,k2: \n"; cin >> k1 >> k2;  for (j = 0; j<n; j++)  swap(x[k1][j], x[k2][j]);  cout << endl;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,i,j, k1, k2 int  x := [10][10]int{}  fmt.Print("m(<10)")  fmt.Scan(&m)  for i=0; i<m; i++ {  for j=0; j<m; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  fmt.Print("k1, k2(<10)")  fmt.Scan(&k1, &k2)  for j:=0; j<m; j++{  x[k1][j], x[k2][j]=x[k2][j], x[k1][j]  }  for i=0; i<m; i++ {  for j=0; j<m; j++ {  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  } | |

1. Дана матрица размера M×N и целое число *K* (1 ≤ *K* ≤ *M*). Удалить строку матрицы с номером *K*.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array** [1..10, 1..10] **of** integer;  K, M, N, i, j: Integer;  **begin**  Write('N,M,K: ');  Readln(N, M, K);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i,j]:=Random(20);  write(a[i,j],'');  **end**;  writeln;  **end**;  writeln;  M := M - 1;  **for** i := K **to** M **do**  **for** j := 1 **to** N **do**  a[i,j]:=a[i+1,j];  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  Write(a[i,j],'');  **end**;  writeln();  **end**;  **end**. | #include <iostream>  using namespace std;  int main() {  int \*\*x, m, n, i, j, k;  cin >> m >> n;  x = new int\*[m];  for (i = 0; i < m; i++)  x[i] = new int[n];  for (i = 0; i < m; i++)  for (j = 0; j < n; j++)  x[i][j] = rand() % 50;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  cout << "k="; cin >> k;  m = m - 1;  for (i = k - 1; i < m; i++)  for (j = 0; j < n; j++)  x[i][j] = x[i + 1][j];  cout << endl;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,n,i,j, k int  x := [10][10]int{}  fmt.Print("m, n(<10)")  fmt.Scan(&m, &n)  for i=0; i<m; i++ {  for j=0; j<n; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  fmt.Print("k(<10)")  fmt.Scan(&k)  m=m-1  for i=k; i<m; i++ {  for j:=0; j<n; j++{  x[i][j]=x[i+1][j]  }  }  for i=0; i<m; i++ {  for j=0; j<n; j++ {  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  } | |

1. Дана матрица размера M×N и целое число *K* (1 ≤ *K* ≤ *M*). Перед строкой матрицы с номером *K* вставить строку из нулей.

| PascalABC.NET |
| --- |
| **Var**  a: **array** [1..10, 1..10] **of** integer;  K, M, N, i, j: Integer;  **begin**  Write('N,M,K: ');  Readln(N, M, K);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i, j] := Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;  **if** k > 0 **then**  **begin**  M := M + 1;  **for** i := M **downto** k + 1 **do**  **for** j := 1 **to** N **do**  a[i, j] := a[i - 1, j];  **end**;  **for** j := 1 **to** N **do**  a[k, j] := 0;  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  Write(a[i, j], ' ');  **end**;  writeln();  **end**;  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main() {  int \*\*x, m, n, i, j, k;  cin >> m >> n;  x = new int\*[m+2];  for (i = 0; i < m+2; i++)  x[i] = new int[n];  for (i = 0; i < m; i++)  for (j = 0; j < n; j++)  x[i][j] = rand() % 50;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  cout << "k="; cin >> k;  k = k - 1;  m = m + 1;  for (i = m - 1; i > k; i--)  for (j = 0; j < n; j++)  x[i][j] = x[i - 1][j];  for (j = 0; j < n; j++)  x[k][j] = 0;  cout << endl;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  system("pause");  return 0;  } |
| Go |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,n,i,j, k int  x := [10][10]int{}  fmt.Print("m, n(<10)")  fmt.Scan(&m, &n)  for i=0; i<m; i++ {  for j=0; j<n; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  fmt.Print("k(<10)")  fmt.Scan(&k)  m=m+1  for i=m-1; i>k; i--{  for j:=0; j<n; j++{  x[i][j]=x[i-1][j]  }  }  for j=0; j<n; j++{  x[k][j]=0  }  for i=0; i<m; i++ {  for j=0; j<n; j++ {  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  } |

1. Дана матрица размера M×N. Элемент матрицы называется ее *локальным минимумом*, если он меньше всех окружающих его элементов. Заменить все локальные минимумы данной матрицы на нули. При решении допускается использовать вспомогательную матрицу.

| PascalABC.NET |
| --- |
| **Type**  matrix = **array** [1..10, 1..10] **of** integer;  **var**  a: matrix;  K, M, N, i, j: Integer;  **function** LocMinMatr(**var** mat: matrix; X, Y: Integer): Boolean;  **begin**  LocMinMatr := true;  **if** x > 1 **then**  **if** mat[x, y] >= mat[x - 1, y] **then**  LocMinMatr := False;  **if** x = mat[x + 1, y] **then**  LocMinMatr := False;  **if** y > 1 **then**  **if** mat[x, y] >= mat[x, y - 1] **then**  LocMinMatr := False;  **if** y = mat[x, y + 1] **then**  LocMinMatr := False;  **if** (y > 1) **and** (x > 1) **then**  **if** mat[x, y] >= mat[x - 1, y - 1] **then**  LocMinMatr := False;  **if** (y < N) **and** (x < M) **then**  **if** mat[x, y] >= mat[x + 1, y + 1] **then**  LocMinMatr := False;  **if** (y > 1) **and** (x < M) **then**  **if** mat[x, y] >= mat[x + 1, y - 1] **then**  LocMinMatr := False;  **if** (y < N) **and** (x > 1) **then**  **if** mat[x, y] >= mat[x - 1, y + 1] **then**  LocMinMatr := False;  **end**;  **begin**  Write('N,M: ');  Readln(N, M);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i, j] := Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;    **for** i := 1 **to** M **do**  **for** j := 1 **to** N **do**  **if** LocMinMatr(a, i, j) **then**  a[i, j] := 0;    **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  Write(a[i, j], ' ');  **end**;  writeln;  **end**;  **end**. |
| C++ |
| #include <iostream>  using namespace std;  int main() {  int \*\*x, m, n, i, j, k;  cin >> m >> n;  x = new int\*[m+2];  for (i = 0; i < m+2; i++)  x[i] = new int[n];  for (i = 0; i < m; i++)  for (j = 0; j < n; j++)  x[i][j] = rand() % 50;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  int num;  for (i = 0; i < m; ++i) {  for (j = 0; j < n; ++j) {  num = 0;  if (i - 1 >= 0) {  if (x[i][j] < x[i - 1][j]) num++;  }  else ++num;  if ((i + 1) < m) {  if (x[i][j] < x[i + 1][j]) num++;  }  else ++num;  if (j - 1 >= 0) {  if (x[i][j] < x[i][j - 1]) num++;  }  else ++num;  if ((j + 1) < n) {  if (x[i][j] < x[i][j + 1]) num++;  }  else  ++num;  if (num == 4) x[i][j] = 0;  }  }  cout << endl;  for (i = 0; i < m; i++) {  for (j = 0; j < n; j++)  cout << x[i][j] << " ";  cout << endl;  }  for (i = 0; i < m + 2; i++)  delete x[i];  delete x;  system("pause");  return 0;  } |

## **Диагонали квадратной матрицы**

1. Дана квадратная матрица *A* порядка N. Найти сумму элементов ее *главной диагонали*, то есть диагонали, содержащей следующие элементы:

*A*1,1,    *A*2,2,    *A*3,3,    …,    *AM*,*M*.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a: **array** [1..10, 1..10] **of**  integer;  N,i,j: Integer;  **begin**  Write('N ');  Readln(N);  **for** i := 1 **to** N **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i, j] := Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;  **var** S:=0;  **for** i:=1 **to** N **do**  S+=a[i,i];  WriteFormat('Sum={0}',S);  **end**. | #include <iostream>  using namespace std;  int main() {  int \*\*x, m, i, j, S = 0;  cin >> m;  x = new int\*[m];  for (i = 0; i < m; i++)  x[i] = new int[m];  for (i = 0; i < m; i++)  for (j = 0; j < m; j++)  x[i][j] = rand() % 50;  for (i = 0; i < m; i++) {  for (j = 0; j < m; j++)  cout << x[i][j] << " ";  cout << endl;  }  for (i = 0; i < m; i++)  S += x[i][i];  cout << S << endl;  for (i = 0; i < m; i++)  delete x[i];  delete x;  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,i,j int  x := [10][10]int{}  fmt.Print("m, n(<10)")  fmt.Scan(&m)  for i=0; i<m; i++ {  for j=0; j<m; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  S:=0  for j=0; j<m; j++{  S+=x[j][j]  }  fmt.Println(S)  } | |

1. Дана квадратная матрица *A* порядка *N*. Найти сумму элементов каждой ее диагонали, параллельной главной (начиная с одноэлементной диагонали *A*1,*M*).

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  a:**array** [1..10,1..10] **of** integer;  S,N, i, j:Integer;    **Begin**  Write('N ');  Readln(N);  **for** i := 1 **to** N **do**  **begin**  **for** j := 1 **to** N **do**  **begin**  a[i,j]:=Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;    **for** j:=1 **to** N-1 **do**  **begin**  S:=0;  **for** i:=1 **to** j **do**  **begin**  S:=S+a[i,N-(j-i)];  **end**;  Writeln(S);  **end**;    **for** j:=N-1 **downto** 1 **do**  **begin**  S:=0;  **for** i:=1 **to** j **do**  **begin**  S:=S+a[N-(j-i),i];  **end**;  Writeln(S);  **end**;  **end**. | #include <iostream>  using namespace std;  int main() {  int \*\*x, m, i, j, S = 0;  cin >> m;  x = new int\*[m];  for (i = 0; i < m; i++)  x[i] = new int[m];  for (i = 0; i < m; i++)  for (j = 0; j < m; j++)  x[i][j] = rand() % 50;  for (i = 0; i < m; i++) {  for (j = 0; j < m; j++)  cout << x[i][j] << " ";  cout << endl;  }  m--;  for (j = 0; j <= m; j++) {  S = 0;  for (i = 0; i <= j; i++)  S += x[i][m - (j - i)];  cout << "S=" << S << endl;  }  for (j = m - 1; j >= 0; j--) {  S = 0;  for (i = 0; i <= j; i++)  S += x[m - (j - i)][i];  cout << "S=" << S << endl;  }  for (i = 0; i < m; i++)  delete x[i];  delete x;  system("pause");  return 0;  } |
| Go | |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,i,j int  x := [10][10]int{}  fmt.Print("m(<10)")  fmt.Scan(&m)  for i=0; i<m; i++ {  for j=0; j<m; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  m--  for j=0; j<=m; j++{  S:=0  for i=0; i<=j; i++{  S+=x[i][m-(j-i)]  }  fmt.Println(S)  }  for j=m-1; j>=0; j--{  S:=0  for i=0; i<=j; i++{  S+=x[m-(j-i)][i]  }  fmt.Println(S)  }  } | |

1. Дана квадратная матрица *A* порядка *M*. Зеркально отразить ее элементы относительно побочной диагонали. (при этом элементы побочной диагонали останутся на прежнем месте, элемент *A*1,1 поменяется местами с *AM*,*M*, элемент *A*1,2 — с *AM*–1,*M* и т. д.). Вспомогательную матрицу не использовать.

| PascalABC.NET |
| --- |
| **Type**  matrix = **array** [1..10, 1..10] **of** integer;  **var**  a: matrix;  M, i, j: Integer;  **begin**  Write('M: ');  Readln(M);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** M **do**  **begin**  a[i, j] := Random(20);  write(a[i, j], ' ');  **end**;  writeln;  **end**;  writeln;  **for** j := 1 **to** M **do**  **for** i := 1 **to** M - j **do**  Swap(a[i, j], a[M - j + 1, M - i + 1]);  **for** i := 1 **to** M **do**  **begin**  **for** j := 1 **to** M **do**  **begin**  Write(a[i,j],' ');  **end**;  writeln();  **end**;  **end**. |
| C++ |
| #include <iostream>  using namespace std;  void swap(int &a, int &b) {  int h = a;  a = b;  b = h;  }  int main() {  int \*\*x, m, i, j, S = 0;  cin >> m;  x = new int\*[m];  for (i = 0; i < m; i++)  x[i] = new int[m];  for (i = 0; i < m; i++)  for (j = 0; j < m; j++)  x[i][j] = rand() % 50;  for (i = 0; i < m; i++) {  for (j = 0; j < m; j++)  cout << x[i][j] << " ";  cout << endl;  }  m--;  for (j = 0; j <= m; j++)  for (i = 0; i <= m - j; i++)  swap(x[i][j], x[m - j][m - i]);  cout << endl;  m++;  for (i = 0; i < m; i++) {  for (j = 0; j < m; j++)  cout << x[i][j] << " ";  cout << endl;  }  for (i = 0; i < m; i++)  delete x[i];  delete x;  system("pause");  return 0;  } |
| Go |
| package main  import (  "fmt"  "math/rand"  )  func main() {  var m,i,j int  x := [10][10]int{}  fmt.Print("m(<10)")  fmt.Scan(&m)  for i=0; i<m; i++ {  for j=0; j<m; j++ {  x[i][j] = rand.Intn(20)  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  fmt.Println()  m--  for j=0; j<=m; j++{  for i=0; i<=m-j; i++{  x[i][j],x[m-j][m-i]=x[m-j][m-i],x[i][j]  }  }  for i=0; i<=m; i++ {  for j=0; j<=m; j++ {  fmt.Printf("%+v ", x[i][j])  }  fmt.Println()  }  } |

**Задачи для самостоятельной работы**

**Вариант 1**

1. Дан целочисленный массив *A* размера *N*. Переписать в новый целочисленный массив *B* того же размера вначале все элементы исходного массива с четными номерами, а затем — с нечетными:

*A*2,    *A*4,    *A*6,    …,    *A*1,    *A*3,    *A*5,    … .

Условный оператор не использовать.

1. Дан массив размера *N*. Поменять порядок его элементов на обратный.
2. Дан массив размера N. Найти номер его последнего локального максимума (локальный максимум — это элемент, который больше любого из своих соседей).
3. Дан целочисленный двумерный массив, размерности n х m. Найти сумму и произведение всех элементов массива.

**Вариант 2**

1. Найти сумму и произведение первых пяти элементов массива.
2. Дано целое число *N* (> 0). Сформировать и вывести целочисленный массив размера *N*, содержащий степени двойки от первой до *N*-й: 2, 4, 8, 16, … .
3. Дан массив *A* размера *N*. Сформировать новый массив *B* того же размера по следующему правилу: элемент *BK* равен сумме элементов массива *A* с номерами от 1 до *K*.
4. Дан целочисленный двумерный массив, размерности n х m. Найти сумму и произведение четных элементов.

**Вариант 3**

1. Найти сумму элементов с k1-го по k2-ой, где k1 и k2 вводятся с клавиатуры. Сделайте проверку корректности их ввода.
2. Дано целое число *N* (> 1), а также первый член *A* и разность *D* *арифметической прогрессии*. Сформировать и вывести массив размера *N*, содержащий *N* первых членов данной прогрессии:

*A*,    *A* + *D*,    *A* + 2·*D*,    *A* + 3·*D*,    … .

1. Дан массив размера *N*. Переставить в обратном порядке элементы массива, расположенные между его минимальным и максимальным элементами, включая минимальный и максимальный элементы.
2. Дан целочисленный двумерный массив, размерности n х m. Найти сумму и произведение элементов, кратных 3 и 5.

**Вариант 4**

1. Найти сумму элементов, больших данного числа А (А вводится с клавиатуры).
2. Дано целое число *N* (> 1), а также первый член *A* и знаменатель *Q* *геометрической прогрессии*. Сформировать и вывести массив размера *N*, содержащий *N* первых членов данной прогрессии:

*A*,    *A*·*D*,    *A*·*D*2,    *A*·*D*3,    … .

1. Дан массив размера *N*. Обнулить все его *локальные максимумы* (то есть числа, большие своих соседей).
2. Дан целочисленный двумерный массив, размерности n х m. Найти количество отрицательных элементов, больше -9.

**Вариант 5**

1. Найти сумму элементов, принадлежащих промежутку от А до В (А и В вводятся с клавиатуры).
2. Дано целое число *N* (> 2). Сформировать и вывести целочисленный массив размера *N*, содержащий *N* первых элементов последовательности *чисел Фибоначчи* *FK*:

*F*1 = 1,        *F*2 = 1,        *FK* = *FK*–2 + *FK*–1,    *K* = 3, 4, … .

1. Дан массив размера *N*. Возвести в квадрат все его *локальные минимумы* (то есть числа, меньшие своих соседей).
2. Дан целочисленный двумерный массив, размерности n х m. Найти номера нечетных элементов, стоящих на четных местах.

**Вариант 6**

1. Найти номера всех отрицательных элементов (вывести их на экран), если таких нет, то сообщить об этом.
2. Даны целые числа *N* (> 2), *A* и *B*. Сформировать и вывести целочисленный массив размера *N*, первый элемент которого равен *A*, второй равен *B*, а каждый последующий элемент равен сумме всех предыдущих.
3. Дан массив размера *N*. Заменить каждый элемент массива на среднее арифметическое этого элемента и его соседей.
4. Дан целочисленный двумерный массив, размерности n х m. Найти максимум и минимум. Поменять их местами.

**Вариант 7**

1. Найти номера всех элементов с максимальным значением.
2. Дан массив размера *N*. Вывести его элементы в обратном порядке.
3. Дан массив размера *N*. Осуществить *сдвиг* элементов массива вправо на одну позицию (при этом *A*1 перейдет в *A*2, *A*2 — в *A*3, …, *AN*–1 — в *AN*, a исходное значение последнего элемента будет потеряно). Первый элемент полученного массива положить равным 0.
4. Дан целочисленный двумерный массив, размерности n х m. Заменить все элементы на их квадраты.

**Вариант 8**

1. Найти количество и произведение отрицательных элементов.
2. Дан целочисленный массив размера *N*. Вывести все содержащиеся в данном массиве нечетные числа в порядке возрастания их индексов, а также их количество *K*.
3. Дан массив размера *N*. Осуществить *сдвиг* элементов массива влево на одну позицию (при этом *AN* перейдет в *AN*–1, *AN*–1 — в *AN*–2, …, *A*2 — в *A*1, a исходное значение первого элемента будет потеряно). Последний элемент полученного массива положить равным 0.
4. Дан целочисленный двумерный массив, размерности n х m. Найти среднее арифметическое всех элементов массива.

**Вариант 9**

1. Найти количество и сумму нечетных элементов.
2. Дан целочисленный массив размера *N*. Вывести все содержащиеся в данном массиве четные числа в порядке убывания их индексов, а также их количество *K*.
3. Дан массив размера *N* и целое число *K* (1<=*K* < *N*). Осуществить *сдвиг* элементов массива вправо на *K* позиций (при этом *A*1 перейдет в *AK*+1, *A*2 — в *AK*+2, …, *AN*–*K* — в *AN*, а исходное значение *K* последних элементов будет потеряно). Первые *K* элементов полученного массива положить равными 0.
4. Дан целочисленный двумерный массив, размерности n х m. Выяснить, какое число встречается в какой строке раньше — положительное или отрицательное.

**Вариант 10**

1. Сколько элементов массива превосходят по модулю заданное число А.
2. Дан целочисленный массив размера *N*. Вывести вначале все содержащиеся в данном массиве четные числа в порядке возрастания их индексов, а затем — все нечетные числа в порядке убывания их индексов.
3. 8) Дан массив размера *N* и целое число *K* (1 <=*K* < *N*). Осуществить *сдвиг* элементов массива влево на *K* позиций (при этом *AN* перейдет в *AN*–*K*, *AN*–1 — в *AN*–*K*–1, …, *AK*+1 — в *A*1, а исходное значение *K* первых элементов будет потеряно). Последние *K* элементов полученного массива положить равными 0.
4. Дан целочисленный двумерный массив, размерности n х m. Выяснить, в какой строке последовательность является возрастающей или убывающей.

**Вариант 11**

1. Найти все элементы, кратные 3 или 5. Сколько их?
2. Дан массив *A* размера *N* и целое число *K* (1 <= *K* <= *N*). Вывести элементы массива с порядковыми номерами, кратными *K*:

*AK*, *A*2·*K*, *A*3·*K*, … .

Условный оператор не использовать.

1. Дан массив размера *N*. Осуществить *циклический сдвиг* элементов массива вправо на одну позицию (при этом *A*1 перейдет в *A*2, *A*2 — в *A*3, …, *AN* — в *A*1).
2. Дан целочисленный двумерный массив, размерности n х m. Вывести его элементы, индексы которых являются степенями двойки (1, 2, 4, 8, 16, ...).

**Вариант 12**

1. Есть ли в данном массиве два соседних положительных элемента? Найти номера первой (последней) пары.
2. Дан массив *A* размера *N* (*N* — четное число). Вывести его элементы с четными номерами в порядке возрастания номеров: *A*2, *A*4, *A*6, …, *AN*. Условный оператор не использовать.
3. Дан массив размера *N*. Осуществить *циклический сдвиг* элементов массива влево на одну позицию (при этом *AN* перейдет в *AN*–1, *AN*–1 — в *AN*–2, …, *A*1 — в *AN*).
4. Дан целочисленный двумерный массив, размерности n х m. Найти количество элементов кратных 7.

**Вариант 13**

1. Есть ли в данном массиве элемент, равный заданному числу? Если есть, то вывести номер одного из них.
2. Дан массив *A* размера *N* (*N* — нечетное число). Вывести его элементы с нечетными номерами в порядке убывания номеров: *AN*, *AN*–2, *AN*–4, …, *A*1. Условный оператор не использовать.
3. Дан массив *A* размера *N* и целое число *K* (1  *K*  4, *K* < *N*). Осуществить *циклический сдвиг* элементов массива вправо на *K* позиций (при этом *A*1 перейдет в *AK*+1, *A*2 — в *AK*+2, …, *AN* — в *AK*). Допускается использовать вспомогательный массив из 4 элементов.
4. Дан целочисленный двумерный массив, размерности n х m. Вывести на экран элементы, которые являются квадратами какого-либо числа.

**Вариант 14**

1. Дан одномерный массив. Переставить в обратном порядке элементы массива, расположенные между минимальным и максимальным элементами.
2. Дан массив *A* размера *N*. Вывести вначале его элементы с четными номерами (в порядке возрастания номеров), а затем — элементы с нечетными номерами (также в порядке возрастания номеров):

*A*2,    *A*4,    *A*6,    …,    *A*1,    *A*3,    *A*5,    … .

Условный оператор не использовать.

1. Дан массив *A* размера *N* и целое число *K* (1 <= *K* <= 4, *K* < *N*). Осуществить *циклический сдвиг* элементов массива влево на *K* позиций (при этом *AN* перейдет в *AN*–*K*, *AN*–1 — в *AN*–*K*–1, …, *A*1 — в *AN*–*K*+1). Допускается использовать вспомогательный массив из 4 элементов.
2. Дан целочисленный двумерный массив, размерности n х m. Найти номера нечетных элементов, стоящих на четных местах.

**Вариант 15**

1. Дан одномерный массив В, состоящий из 2n элементов. Переставить его элементы по следующему правилу: b[n+1], b[n+2],..., b[2n], b[1], b[2],..., b[n].
2. Дан массив *A* размера *N*. Вывести вначале его элементы с нечетными номерами в порядке возрастания номеров, а затем — элементы с четными номерами в порядке убывания номеров.

*A*1,    *A*3,    *A*5,    …,    *A*6,    *A*4,    *A*2.

Условный оператор не использовать.

1. Дан массив размера *N*, все элементы которого, кроме первого, упорядочены по возрастанию. Сделать массив упорядоченным, переместив первый элемент на новую позицию.
2. Дан целочисленный двумерный массив, размерности n х m. Найти максимум и минимум. Поменять их местами.

**Вариант 16**

1. Дан одномерный массив В, состоящий из 2n элементов. Переставить его элементы по следующему правилу: b[n+1] b[n+2],..., b[2n], b[n-1],..., b[1].
2. Дан массив *A* размера *N*. Вывести его элементы в следующем порядке:

*A*1,    *AN*,    *A*2,    *AN*–1,    *A*3,    *AN*–2,    … .

1. Дан массив размера *N*, все элементы которого, кроме последнего, упорядочены по возрастанию. Сделать массив упорядоченным, переместив последний элемент на новую позицию.
2. Дан целочисленный двумерный массив, размерности n х m. Заменить все элементы на их квадраты.

**Вариант 17**

1. Дан одномерный массив В, состоящий из 2n элементов. Переставить его элементы по следующему правилу: b[1], b[n+1], b[2], b[n+2],..., b[n], b[2n].
2. Дан массив *A* размера *N*. Вывести его элементы в следующем порядке:

*A*1,    *A*2,    *AN*,    *AN*–1,    *A*3,    *A*4,    *AN*–2,    *AN*–3,    … .

1. Дан массив размера *N*, все элементы которого, кроме одного, упорядочены по убыванию. Сделать массив упорядоченным, переместив элемент, нарушающий упорядоченность, на новую позицию.
2. Дан целочисленный двумерный массив, размерности n х m. Заменить все элементы на их противоположные значения.

**Вариант 18**

1. Дан одномерный массив В, состоящий из 2n элементов. Переставить его элементы по следующему правилу: b[2n], b[2n-1],..., b[n+1], b[1], b[2],..., b[n].
2. Дан массив размера *N*. Найти максимальный из его локальных минимумов.
3. Дан массив размера *N* и целое число *K* (1 <= K <= N). Удалить из массива элемент с порядковым номером *K*.
4. Дан целочисленный двумерный массив, размерности n х m. Поменять местами первый и последний элементы.

**Вариант 19**

1. Дан одномерный массив В, состоящий из 2n элементов. Переставить его элементы по следующему правилу: первый и последний отрицательный.
2. Дан массив размера *N*. Найти минимальный из его локальных максимумов (определение *локального максимума* дано в задании Array33).
3. Дан массив размера *N* и целые числа *K* и *L* (1 <= *K* < *L* <= *N*). Удалить из массива элементы с номерами от *K* до *L* включительно и вывести размер полученного массива и его содержимое.
4. Дан целочисленный двумерный массив, размерности n х m. Сформировать новый массив, состоящий из противоположных соответствующих элементов .

**Вариант 20**

1. Дан одномерный массив В, состоящий из 2n элементов. Переставить его элементы по следующему правилу: второй и минимальный.
2. Дан массив размера *N*. Найти максимальный из его элементов, не являющихся ни локальным минимумом, ни локальным максимумом. Если таких элементов в массиве нет, то вывести 0.
3. Дан целочисленный массив размера *N*. Удалить из массива все нечетные числа и вывести размер полученного массива и его содержимое.
4. Дан целочисленный двумерный массив, размерности n х m. Вывести на экран те элементы, у которых остаток от деления на m равен k.

**Вариант 21**

1. Найти сумму всех чётных элементов массива, стоящих на чётных местах, то есть имеющие чётные номера.
2. Дан массив размера *N*. Найти количество участков, на которых его элементы монотонно возрастают.
3. Дан целочисленный массив размера *N* (> 2). Удалить из массива все элементы с четными номерами (2, 4, …). Условный оператор не использовать.
4. Вводятся результаты контрольной работы 10 учащихся. Определите число не удовлетворительных, удовлетворительных, хороших, отличных оценок. Вывести среднюю оценку, полученную учащимися за контрольную работу.

**Вариант 22**

1. Найти сумму и произведение первых пяти элементов массива.
2. Дан массив размера *N*. Найти количество участков, на которых его элементы монотонно убывают.
3. Дан целочисленный массив размера *N* (> 2). Удалить из массива все элементы с нечетными номерами (1, 3, …). Условный оператор не использовать.
4. Ввести оценки N учеников по K предметам. Определить и вывести на экран количество учеников, не получивших ни одной "5".

**Вариант 23**

1. Найти сумму элементов с k1-го по k2-ой, где k1 и k2 вводятся с клавиатуры. Сделайте проверку корректности их ввода.
2. Дан массив размера *N*. Найти количество его *промежутков монотонности* (то есть участков, на которых его элементы возрастают или убывают).
3. Дан целочисленный массив размера *N*. Удалить из массива все соседние одинаковые элементы, оставив их первые вхождения.
4. В группе учится N студентов, студенты получили по четыре отметки за экзамен. Определить количество неуспевающих студентов и средний балл группы.

**Вариант 24**

1. Найти сумму элементов, больших данного числа А (А вводится с клавиатуры).
2. Дано число *R* и массив *A* размера *N*. Найти элемент массива, который *наиболее близок* к числу *R* (то есть такой элемент *AK*, для которого величина |*AK* – *R*| является минимальной).
3. Дан целочисленный массив размера *N*. Удалить из массива все одинаковые элементы, оставив их первые вхождения.
4. Дан целочисленный двумерный массив, размерности n х m. Вычислить сумму чисел, порядковые номера которых являются числами фибоначчи.

**Вариант 25**

1. Дан массив *A* ненулевых целых чисел размера 10. Вывести значение первого из тех его элементов *AK*, которые удовлетворяют неравенству *AK* < *A*10. Если таких элементов нет, то вывести 0.
2. Дан массив размера *N*. Найти два соседних элемента, сумма которых максимальна, и вывести эти элементы в порядке возрастания их индексов.
3. Дан целочисленный массив размера *N*. Удалить из массива все одинаковые элементы, оставив их последние вхождения.
4. Дан целочисленный двумерный массив, размерности n х m. Сложить соответствующие элементы.

**Вариант 26**

1. Дан целочисленный массив *A* размера 10. Вывести порядковый номер последнего из тех его элементов *AK*, которые удовлетворяют двойному неравенству *A*1 < *AK* < *A*10. Если таких элементов нет, то вывести 0.
2. Дано число *R* и массив размера *N*. Найти два соседних элемента массива, сумма которых наиболее близка к числу *R*, и вывести эти элементы в порядке возрастания их индексов (определение наиболее близких чисел дано в задании Array40).
3. Дан целочисленный массив размера *N*. Удалить из массива все элементы, встречающиеся менее трех раз, и вывести размер полученного массива и его содержимое.
4. Дана действительная квадратная матрица порядка n. Найти наибольшее из значений элементов, расположенных в заштрихованной части матрицы.

а) б) в) г)

http://valera.asf.ru/delphi/lab/1/image082.gif

**Вариант 27**

1. Дан массив размера *N* и целые числа *K* и *L* (1 <= *K* <= *L* <= *N*). Найти сумму элементов массива с номерами от *K* до *L* включительно.
2. Дан целочисленный массив размера *N*, все элементы которого упорядочены (по возрастанию или по убыванию). Найти количество различных элементов в данном массиве.
3. Дан целочисленный массив размера *N*. Удалить из массива все элементы, встречающиеся более двух раз, и вывести размер полученного массива и его содержимое.
4. Дана квадратная вещественная матрица размерности n. Найти количество нулевых элементов, стоящих: выше главной диагонали; ниже главной диагонали; выше и ниже побочной.

**Вариант 28**

1. Дан массив размера *N* и целые числа *K* и *L* (1<=*K <= L <= N*). Найти среднее арифметическое элементов массива с номерами от *K* до *L* включительно.
2. Дан целочисленный массив размера *N*, содержащий ровно два одинаковых элемента. Найти номера одинаковых элементов и вывести эти номера в порядке возрастания.
3. Дан целочисленный массив размера *N*. Удалить из массива все элементы, встречающиеся ровно два раза, и вывести размер полученного массива и его содержимое.
4. Дана вещественная матрица размерности n \* m. По матрице получить логический вектор, присвоив его k-ому элементу значение True , если выполнено указанное условие и значение False иначе: - все элементы k столбца нулевые; - элементы k строки матрицы упорядочены по убыванию; - k строка массива симметрична.

**Вариант 29**

1. Дан массив размера *N* и целые числа *K* и *L* (1 < *K <= L <= N*). Найти сумму всех элементов массива, кроме элементов с номерами от *K* до *L* включительно.
2. Дан массив размера *N*. Найти номера двух ближайших элементов из этого массива (то есть элементов с наименьшим модулем разности) и вывести эти номера в порядке возрастания.
3. Дан массив размера *N* и целое число *K* (1 <= K <= N). Перед элементом массива с порядковым номером *K* вставить новый элемент с нулевым значением.
4. Дана вещественная матрица размерности n \* m. Сформировать вектор b, в котором элементы вычисляются как: - произведение элементов соответствующих строк; - среднее арифметическое соответствующих столбцов; - разность наибольших и наименьших элементов соответствующих строк; - значения первых отрицательных элементов в столбце.

**Вариант 30**

1. Дан массив размера *N* и целые числа *K* и *L* (1 < *K <= L <= N*). Найти среднее арифметическое всех элементов массива, кроме элементов с номерами от *K* до *L* включительно.
2. Дано число *R* и массив размера *N*. Найти два различных элемента массива, сумма которых наиболее близка к числу *R*, и вывести эти элементы в порядке возрастания их индексов
3. Дан массив *A* размера *N*. Сформировать новый массив *B* того же размера по следующему правилу: элемент *BK* равен среднему арифметическому элементов массива *A* с номерами от 1 до *K*.
4. Дан двухмерный массив A[1..m,1..n]. Написать программу построения одномерного массива B[1..m], элементы которого соответственно равны а) суммам элементов строк, б) произведениям элементов строк, в) наименьшим средних арифметических элементов строк.

**Вариант 31**

1. Дан целочисленный массив размера *N*, не содержащий одинаковых чисел. Проверить, образуют ли его элементы *арифметическую прогрессию*. Если образуют, то вывести разность прогрессии, если нет — вывести 0.
2. Дан целочисленный массив размера *N*. Найти количество различных элементов в данном массиве.
3. Дан массив *A* размера *N*. Сформировать новый массив *B* того же размера по следующему правилу: элемент *BK* равен сумме элементов массива *A* с номерами от *K* до *N*.
4. Расположить элементы данного массива в обратном порядке (первый элемент меняется с последним, второй - с предпоследним и т.д. до середины; если массив содержит нечетное количество элементов, то средний остается без изменения).

**Вариант 32**

1. Дан массив ненулевых целых чисел размера *N*. Проверить, образуют ли его элементы *геометрическую прогрессию*. Если образуют, то вывести знаменатель прогрессии, если нет — вывести 0.
2. Дан целочисленный массив размера *N*. Найти максимальное количество его одинаковых элементов.
3. Дан массив *A* размера *N*. Сформировать новый массив *B* того же размера по следующему правилу: элемент *BK* равен среднему арифметическому элементов массива *A* с номерами от *K* до *N*.
4. В данном массиве поменять местами элементы, стоящие на нечетных местах, с элементами, стоящими на четных местах.

**Вариант 33**

1. Дан целочисленный массив размера *N*. Проверить, чередуются ли в нем четные и нечетные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.
2. Дан целочисленный массив размера *N*. Если он является *перестановкой*, то есть содержит все числа от 1 до *N*, то вывести 0; в противном случае вывести номер первого недопустимого элемента.
3. Дан массив *A* размера *N*. Сформировать два новых массива *B* и *C*: в массив *B* записать все положительные элементы массива *A*, в массив *C* — все отрицательные (сохраняя исходный порядок следования элементов). Вывести вначале размер и содержимое массива *B*, а затем — размер и содержимое массива *C*.
4. В массиве А[1..N,1..N] определить номера строки и столбца какой-нибудь седловой точки. Некоторый элемент массива называется седловой точкой, если он является одновременно наименьшим в своей строке и наибольшим в своем столбце.

**Вариант 34**

1. Дан массив ненулевых целых чисел размера *N*. Проверить, чередуются ли в нем положительные и отрицательные числа. Если чередуются, то вывести 0, если нет, то вывести порядковый номер первого элемента, нарушающего закономерность.
2. Дан целочисленный массив *A* размера *N*, являющийся перестановкой (определение *перестановки* дано в задании Array49). Найти количество *инверсий* в данной перестановке, то есть таких пар элементов *AI* и *AJ*, в которых большее число находится слева от меньшего: *AI* > *AJ* при *I* < *J*.
3. Даны два массива *A* и *B* размера 5, элементы которых упорядочены по возрастанию. Объединить эти массивы так, чтобы результирующий массив *C* (размера 10) остался упорядоченным по возрастанию.
4. Массив А[1..5,1..7] содержит вещественные числа. Требуется ввести целое число K и вычислить сумму элементов А[I,J], для которых I+J=К. Прежде, однако следует убедиться, что значение К позволяет найти решение, в противном случае нужно напечатать сообщение об ошибке.

**Вариант 35**

1. Дан массив *A* размера *N*. Найти минимальный элемент из его элементов с четными номерами: *A*2, *A*4, *A*6, … .
2. Даны массивы *A* и *B* одинакового размера *N*. Поменять местами их содержимое и вывести вначале элементы преобразованного массива A, а затем — элементы преобразованного массива B.
3. Даны три целочисленных массива A, B и C размера NA, NB, NC соответственно, элементы которых упорядочены по убыванию. Объединить эти массивы так, чтобы результирующий целочисленный массив D (размера   
   NA + NB + NC) остался упорядоченным по убыванию.
4. Дан массив A[1..N,1..N]. Составить программу, которая прибавила бы каждому элементу данной строки элемент, принадлежащий этой строке и главной диагонали.

**Вариант 36**

1. Дан массив *A* размера *N*. Найти максимальный элемент из его элементов с нечетными номерами: *A*1, *A*3, *A*5, … .
2. Дан массив *A* размера *N*. Сформировать новый массив *B* того же размера, элементы которого определяются следующим образом:

| *BK* | = | 2·*AK*, | если *AK* < 5, |
| --- | --- | --- | --- |
| *AK*/2 | в противном случае. |

1. Дан массив *A* размера *N* и целое число *K* (1 <= K <= N). Преобразовать массив, увеличив каждый его элемент на исходное значение элемента *AK*.
2. Дана матрица NxM. Переставляя ее строки и столбцы, переместить наибольший элемент в верхний левый угол. Определить можно ли таким же образом поместить минимальный элемент в нижний правый угол.:

**Вариант 37**

1. Дан массив размера *N*. Найти номера тех элементов массива, которые больше своего правого соседа, и количество таких элементов. Найденные номера выводить в порядке их возрастания.
2. Даны два массива *A* и *B* одинакового размера *N*. Сформировать новый массив *C* того же размера, каждый элемент которого равен максимальному из элементов массивов *A* и *B* с тем же индексом. Дан целочисленный массив размера *N*.
3. Увеличить все четные числа, содержащиеся в массиве, на исходное значение первого четного числа. Если четные числа в массиве отсутствуют, то оставить массив без изменений.
4. Получить матрицу А (m\*n) образованную по соответствующему закону (размер матрицы и закон, по которому определяется каждый элемент матрицы А, для каждого варианта задания приведены в столбце 2 таблицы).

**Вариант 38**

1. Дан массив размера *N*. Найти номера тех элементов массива, которые больше своего левого соседа, и количество таких элементов. Найденные номера выводить в порядке их убывания.
2. Дан целочисленный массив *A* размера *N*. Переписать в новый целочисленный массив *B* все четные числа из исходного массива (в том же порядке) и вывести размер полученного массива *B* и его содержимое.
3. Дан целочисленный массив размера *N*. Увеличить все нечетные числа, содержащиеся в массиве, на исходное значение последнего нечетного числа. Если нечетные числа в массиве отсутствуют, то оставить массив без изменений.
4. Получить матрицу В, осуществив соответствующие преобразования над  матрицей А (преобразование, которое необходимо сделать над матрицей А, для каждого варианта задания приведено в столбце 3 таблицы).

**Вариант 39**

1. Дан массив размера *N*. Найти номер его первого локального минимума (*локальный минимум* — это элемент, который меньше любого из своих соседей).
2. Дан целочисленный массив *A* размера *N*. Переписать в новый целочисленный массив *B* все элементы с нечетными порядковыми номерами (1, 3, …) и вывести размер полученного массива *B* и его содержимое. Условный оператор не использовать.
3. Дан массив размера *N* (*N* — четное число). Поменять местами его первый элемент со вторым, третий — с четвертым и т. д.
4. Получить матрицу Е, переставив блоки матрицы D (необходимые перестановки для каждого варианта задания приведены в столбце 5 таблицы).

**Вариант 40**

1. Введите одномерный целочисленный массив. Найдите наибольший нечетный элемент. Далее трижды осуществите циклический сдвиг влево элементов, стоящих справа от найденного максимума, и один раз сдвиг элементов вправо, стоящих слева от найденного максимума.
2. Имеются два упорядоченных по возрастанию (предыдущий элемент меньше последующего) массива. Требуется получить третий упорядоченный по возрастанию массив, путем слияния первых двух.
3. Осуществить перестановку элементов одномерного массива без использования дополнительного массива.
4. В матрице удалите столбцы с положительными суммами элементов, а затем в качестве первого вставьте столбец из минимальных элементов соответствующих строк.

# **Символы и строки: группа String**

При выполнении заданий на обработку русских букв можно считать, что буква «ё» в исходных строковых данных отсутствует. Это связано с тем, что расположение данного символа в кодовой таблице не соответствует его расположению в русском алфавите, и поэтому учет символа «ё» во многих алгоритмах требует дополнительных действий.

## **Символы и их коды. Формирование строк**

1. Дан символ *C*. Вывести его *код* (то есть номер в кодовой таблице).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **var**  C: Char;  **begin**  Write('C:');  readln(C);  Writeln('C:', ord(C));  **end**. | | #include <iostream>  #include <windows.h>  using namespace std;  int main()  {  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  unsigned char C;  int kod;  cout << "Введите символ С: ";  cin >> C;  kod = C;  cout << kod << endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| c=input()  print(ord(c)) | static void Main(string[] args)  {  char c = Convert.ToChar(Console.ReadLine());  int m = (int)c;  Console.WriteLine(m);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  char c=sc.nextLine().charAt(0);  int n=(int)c;  System.out.println(n);  } | | | package main  import "fmt"  func main() {  var s string  fmt.Scanln(&s)  var n=int(s[0])  fmt.Println(n)  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** c = reader.nextLine().get(0);  **var** n = c.toInt()  *println*(n) } | | | |

1. Дано целое число *N* (32 ≤ *N* ≤ 126). Вывести символ с кодом, равным *N*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **var**  N: byte;  **begin**  Write('N:');  readln(N);  Writeln('N:', chr(N));  **end**. | | #include <iostream>  #include <windows.h>  using namespace std;  int main()  {  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  unsigned char C;  int n;  cout << "Введите символ С: ";  cin >> n;  C=n;  cout<<C<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| n=int(input())  print(chr(n)) | static void Main(string[] args)  {  int m=Convert.ToInt32(Console.ReadLine());  int c = (char)m;  Console.WriteLine(c);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  int n=sc.nextInt();  char c=(char)n;  System.out.println(c);  } | | | package main  import "fmt"  func main() {  var n int  fmt.Scanln(&n)  ch := rune(n)  fmt.Printf("char: %c\n", ch)  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** n = reader.nextInt()  **var** c= n.toChar()  *println*(c) } | | | |
| VB.NET | | | |
| Sub Main()  Dim c As Char  Dim n As Integer  c = Convert.ToChar(Console.ReadLine())  n = Convert.ToInt32(c)  Console.WriteLine(n)  Console.ReadKey()  End Sub | | | |

1. Дано целое число *N* (1 ≤ *N* ≤ 26). Вывести *N* последних *строчных* (то есть маленьких) букв латинского алфавита в обратном порядке (начиная с буквы «z»).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  N, i: byte;  C: char;  **begin**  Write('N:');  readln(N);  **for** i:=122 **downto** 122-N+1 **do**  Writeln(chr(i));  **end**. | | #include <iostream>  using namespace std;  int main()  {  int n, i;  cin >> n;  for (i = 122; i >= 122 - n; i--)  cout << (char)i << " ";  system("pause");  return 0;  } | |
| Python | C# | | |
| n=int(input())  for i in range (122,122-  n+1,-1):  print(chr(i)); | static void Main(string[] args)  {  int n;  n=Convert.ToInt32(Console.ReadLine());  for (int i = 122; i >= 122 - n; i--)  Console.Write((char)i+" ");  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  int n=sc.nextInt();  for (int i=122; i>=122-n; i--){  char c=(char)i;  System.out.print(c+" ");  } | | | package main  import "fmt"  func main() {  var n int  fmt.Scan(&n)  for i:=122;i>=122-n+1;i--{  ch := rune(i)  fmt.Printf("%c ", ch)  }  } |
| VB.NET | | | |
| Sub Main()  Dim c As Char  Dim n As Integer  n = Convert.ToInt32(Console.ReadLine())  c = Convert.ToChar(n)  Console.WriteLine(c)  Console.ReadKey()  End Sub | | | |

1. Дана непустая строка *S*. Вывести строку, содержащую символы строки *S*, между которыми вставлено по одному пробелу.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **var**  i: integer;  S, S2: string;  **begin**  Write('S:');  Readln(S);  S2 := S[1];  **for** i := 2 **to** Length(S) **do**  S2 := S2 + ' ' + S[i];  Writeln(S2);  **end**. | | #include "stdafx.h"  #include <iostream>  #include <string>  using namespace std;  int main(){  string s, Ss="";  getline(cin, s);  for (int i = 0; i < s.length(); i++){  Ss = Ss+s[i] + " ";  }  cout << Ss;  system("pause");  return 0;  } | |
| Python | C# | | |
|  | static void Main(string[] args){  string s = Console.ReadLine();  int i = 1;  while (i < s.Length){  s = s.Insert(i, " ");  i += 2;  }  Console.WriteLine(s);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  String S1="";  for (int i=0; i<S.length(); i++)  S1=S1+S.charAt(i)+" ";  System.out.print(S1);  } | | | package main  import (  "bufio"  "fmt"  "os"  "strings"  )  func main() {  reader := bufio.NewReader(os.Stdin)  s, \_:=reader.ReadString('\n')  s=strings.Trim(s," ")  var t string  for i:= 0; i < len(s); i++ { t=t+string(s[i])+string("\*");  }  t=strings.Trim(t," \*")  fmt.Printf("%s",t)  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** S= reader.next()  **var** S1=**""  for** (i **in** (0..S.**length**-1))  S1=S1+S.get(i)+**" "**;  *println*(S1) } | | | |

## **Посимвольный анализ и преобразование строк. Строки и числа**

1. Дана строка. Подсчитать количество содержащихся в ней цифр.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  S: string;  **begin**  Write('S:');  Readln(S);    **var** k := 0;  **for var** i:=1 **to** Length(S) **do**  **if** char.IsDigit(S[i]) **then**  k += 1;  Writeln(k);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main(){  string s;  getline(cin, s);  int k = 0, i;  for (i = 0; i < s.length(); i++)  if (isdigit(s[i]))  k++;  cout<< k;  system("pause");  return 0;  } | |
| Python | C# | | |
| S=input("S->")  n=len(S)  k=0  for i in range (0, n):  if S[i].isdigit():  k+=1  print(k) | static void Main(string[] args){  string s = Console.ReadLine();  int k = 0;  for (int i = 0; i < s.Length; i++)  if (Char.IsDigit(s[i]))  k++;  Console.WriteLine(k);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  int i, k=0;  for (i=0; i<S.length(); i++){  int m=(int)S.charAt(i);  if ((m>47) && (m<58))  k++;  }  System.out.println(k);  } | | | package main  import (  "bufio"  "fmt"  "os"  "strings"  )  func main() {  reader:=bufio.NewReader(os.Stdin)  s, \_:=reader.ReadString('\n')  s=strings.Trim(s," ")  var k int=0  for i:=0; i < len(s); i++ {  if int(s[i])>=48 && int(s[i]) <=57{  k++  }  }  fmt.Println(k)  } |

1. Дана строка. Подсчитать общее количество содержащихся в ней строчных латинских и русских букв.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Begin**  **var** S := readstring;  **var** k := 0;  **for var** i:=1 **to** Length(S)**do**  **if** (ord(S[i]) **in** [97..122, 224..239]) **then**  k += 1;  Writeln(k);  **end**. | | #include <iostream>  #include <string>  #include <windows.h>  using namespace std;  bool lower(unsigned char c) {  return (((int)c > 96) && ((int)c <  123)) || (((int)c > 223) &&  ((int)c < 240));  }  int main(){  SetConsoleCP(1251);  SetConsoleOutputCP(1251);  string s;  unsigned char c;  cout<<"Введите символ S: ";  getline(cin, s);  int k = 0, i;  for (i = 0; i < s.length(); i++) {  c = s[i];  if (lower(c))  k++;  }  cout << k;  system("pause");  return 0;  } |
| Python | C# | |
| S=input("S->")  n=len(S)  k=0  for i in range (0, n):  if S[i].isalpha():  k+=1  print(k) | static void Main(string[] args)  {  string s = Console.ReadLine();  int k = 0;  for (int i = 0; i < s.Length; i++)  if (Char.IsUpper(s[i]))  if (s[i]<='Z')  k++;  Console.WriteLine(k);  Console.ReadKey();  } | |
| Go | | |
| package main  import (  "bufio"  "fmt"  "os"  "strings"  )  func IsLetter(s string) int {  var k int=0  for \_, r := range s {  if (r >= 'A' && r <= 'Z') || (r >= 'a' && r <= 'z') || (r >=  'А' && r <= 'Я') || (r >= 'а' && r <= 'я'){  k++;  }  }  return k  }  func main() {  reader := bufio.NewReader(os.Stdin)  s, \_ := reader.ReadString('\n')  s=strings.Trim(s," ")  fmt.Println(IsLetter(s))  } | | |

1. Дано целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке слева направо).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  N: integer;  S: string;  **begin**  Write('N:');  Readln(N);  Str(N, S);  Writeln(S);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main(){  string s;  getline(cin, s);  int m = stoi(s);  cout <<m;  system("pause");  return 0;  } | |
| Python | C# | | |
| S=input("S->")  n=int(S)  print(n) | static void Main(string[] args){  string s = Console.ReadLine();  int m = Convert.ToInt32(s);  Console.WriteLine(m);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  int n=Integer.valueOf(S);  System.out.println(n);  } | | | package main  import (  "fmt"  "strconv"  )  func main() {  var s string  fmt.Scan(&s)  n, \_ := strconv.Atoi(s)  fmt.Println(n)  } |

1. Дана строка, изображающая арифметическое выражение вида «<цифра>±<цифра>±…±<цифра>», где на месте знака операции «±» находится символ «+» или «–» (например, «4+7–2–8»). Вывести значение данного выражения (целое число).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  S: String;  Rez, op,i: integer;  **begin**  Read(S);  Rez := ord(S[1]) - 48;  **for** i := 2 **to** Length(S) **do**  **if** (char.IsDigit(S[i])) **then**  Rez:=Rez+op\*(ord(S[i])-48)  **else if** (ord(S[i]) = 43) **then**  op := 1 //+  **else if** (ord(S[i]) = 45) **then**  op := -1; //-  Writeln(Rez);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main(){  string S;  getline(cin, S);  int Rez = int(S[0]) - 48;  int i, op = 1;  for (i = 1; i < S.size(); i++) {  if (isdigit(S[i]))  Rez = Rez+op\*(int(S[i])-48);  if (int(S[i]) == 43)  op = 1;  else if (int(S[i]) == 45)  op = -1;  }  cout << Rez;  system("pause");  return 0;  } | |
| Python | C# | | |
| S=input("S->")  Rez=ord(S[0])-48  n=len(S)  op=1  for i in range (1, n):  if S[i].isdigit():  Rez=Rez+op\*  (ord(S[i])-48)  elif ord(S[i])==43:  op=1  elif ord(S[i])==45:  op=-1  print(Rez) | static void Main(string[] args){  string s = Console.ReadLine();  int Rez = (int)s[0] - 48;  int op = 1;  for (int i=1; i<s.Length; i++) {  if (s[i] == '+')  op = 1;  else if (s[i]=='-')  op = -1;  else  Rez = Rez + op \* ((int)s[i] - 48);  }  Console.WriteLine(Rez);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args){  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  int op=1, P;  P=(int)S.charAt(0)-48;  for (int i=1; i<S.length(); i++){  int m=(int)S.charAt(i)-48;  if (m==-5)  op=1;  else if (m==-3)  op=-1;  else  P=P+m\*op;  }  System.out.println(P);  } | | | package main  import (  "fmt"  "unicode"  )  func main() {  var S string;  var Rez, op int=0,1  fmt.Scan(&S)  Rez=int(S[0])-48  for i:= 1; i<len(S); i++{  if unicode.IsDigit(rune(S[i])){  Rez=Rez+op\*(int(S[i])-48)  } else if (int(S[i])== 43){  op = 1  } else if (int(S[i])==45){  op= -1;  }  }  fmt.Println(Rez)  } |

## **Обработка строк с помощью стандартных функций. Поиск и замена**

В заданиях, связанных с поиском и заменой подстрок, можно считать, что исходная строка не содержит *перекрывающихся* вхождений требуемых подстрок.

1. Дано целое число *N* (> 0) и строка *S*. Преобразовать строку *S* в строку длины *N* следующим образом: если длина строки *S* больше *N*, то отбросить первые символы, если длина строки *S* меньше *N*, то в ее начало добавить символы «.» (точка).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Begin**  **var** N := ReadInteger;  **var** S := Readstring;  **if** length(S) > N **then**  delete(S,1,length(S)-N)  **else if** length(S) < N **then**  **for var** i:=length(S) **to** N-1 **do**  s := '.' + s;  writeln(S);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main()  {  string S;  getline(cin, S);  int n, i, m;  cin >> n;  m = S.size();  if (m > n)  S=S.substr(m - n, m);  else  S = string(n - m, '\*') + S;  cout << S<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| n=int(input("n->"))  S=input("S->")  S=S.strip() #удалить лишние пробелы  m=len(S)  if m>n:  S=S[m-n:]  else:  for i in range (m,n):  S="."+S  print(S) | static void Main(string[] args) {  string S = Console.ReadLine();  int n=Convert.ToInt32(Console.ReadLine());  int m = S.Length;  if (m > n)  S = S.Substring(m - n, n);  else  S =new string('\*',(n - m)) + S;  Console.WriteLine(S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[]args){  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  int m = S.length();  int n=sc.nextInt();  if (m > n)  S=S.substring(m-n, m);  else  for (int i=0; i<n-m; i++)  S="\*"+S;  System.out.println(S);  } | | | package main  import "fmt"  func main()  {  var S string;  var n,m int  fmt.Scan(&n)  fmt.Scan(&S)  m=len(S)  if m>n{  S= S[:m-n]  }else{  for i:=m; i<n; i++{  S=S+"."  }  }  fmt.Println(S)  } |

1. Дан символ *C* и строки *S*, *S*0. После каждого вхождения символа *C*   
   в строку *S* вставить строку *S*0.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  S, S0: string;  C: Char;  i: integer;  **begin**  Write('C:');  Readln(C);  Write('S:');  Readln(S);  Write('S0:');  Readln(S0);  i := 1;  **while**(i <= length(S)) **do**  **begin**  **if** S[i] = C **then**  **begin**  Insert(S0,S,i+1);  i := i + Length(S0);  **end**;  inc(i);  **end**;  writeln(S);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main(){  string S, S0;  getline(cin, S);  getline(cin, S0);  unsigned char C;  cin >> C;  int i = 0;  while (i < S.size()) {  if (S[i] == C) {  S.insert(i+1, S0);  i += S0.size();  }  else  i++;  }  cout << S << endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| c=input("c->")  S=input("S->")  S0=input("S0->")  #удалить лишние пробелы  S=S.strip()  m=len(S)  p=0  S2=""  for i in range (0,m):  ss=""  if (S[i]==c[0]):  ss=S[p:i+1]+S0  p=i+1  S2=S2+ss  S2=S2+S[p:]  print(S2) | static void Main(string[] args) {  string S = Console.ReadLine();  string S0 = Console.ReadLine();  char c=Convert.ToChar(Console.ReadLine());  int i = 0;  while (i < S.Length) {  if (S[i] == c) {  S=S.Insert(i+1, S0);  i += S0.Length+1;  }  else  i++;  }  Console.WriteLine(S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  String S0=sc.nextLine();  char C=sc.nextLine().charAt(0);  String S1="";  int n=S.length();  int p=0;  for (int i=0; i<n; i++){  if (S.charAt(i)==C){  S1=S1+S.substring(p, i+1)+S0;  p=i+1;  }  }  S1=S1+S.substring(p, S.length());  System.out.println(S1);  } | | | package main  import (  "fmt"  "os"  "bufio"  )  func main() {  reader := bufio.NewReader(os.Stdin)  var C string  fmt.Println("Enter C: ")  fmt.Scanf("%s\n",&C)  fmt.Println("Enter S: ")  S,\_:= reader.ReadString('\n')  fmt.Println("Enter S0: ")  S0,\_:=reader.ReadString('\n')  S2:=""  var i, p int=0,0  for ; i<len(S); i++{  ss:=""  if (S[i]==C[0]){  ss=S[p:i+1]+S0  fmt.Println(ss)  p=i+1  }  S2=S2+ss  }  S2=S2+S[p:]  fmt.Println(S2)  } |

1. Даны строки *S* и *S*0. Удалить из строки *S* последнюю подстроку, совпадающую с *S*0. Если совпадающих подстрок нет, то вывести строку *S* без изменений.

| PascalABC.NET | | | C++ |
| --- | --- | --- | --- |
| **Var**  s, s0: string;  num, poz, newpoz: integer;  **begin**  Write('S: ');  Readln(s);  Write('S0: ');  Readln(s0);  poz := 0;  **repeat**  newpoz:=pos(s0,copy(s,poz+1,length(s)));  **if** newpoz <> 0 **then**  poz := poz + newpoz;  **until** newpoz = 0;  **if** (poz <> 0) **then**  delete(s, poz, length(s0));  Writeln(s);  **end**. | | | #include <iostream>  #include <string>  using namespace std;  int main(){  string S, S0;  getline(cin, S);  getline(cin, S0);  int pos = S.find\_last\_of(S0);  if (pos+S0.size() < S.size())  S = S.substr(0, pos - 2) +  S.substr((pos + S0.size())  - 1, S.size());  else  S = S.substr(0, pos - 2);  cout << S << endl;  system("pause");  return 0;  } |
| Python | C# | | |
| S=input("S->")  S0=input("S0->")  S=S.strip() #удалить лишние пробелы  m=S.rfind(S0) # посл. подстрока  S=S[:m]+S[m+len(S0)+1:]  print(S) | static void Main(string[] args) {  string S = Console.ReadLine();  string S0 = Console.ReadLine();  int pos =S.LastIndexOf(S0);  S=S.Remove(pos, S0.Length);  Console.WriteLine(S);  Console.ReadKey();  } | | |
| Java | | Go | |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  String S0=sc.nextLine();  int pos=S.lastIndexOf(S0);  String S1=S.substring(0, pos)+  S.substring(pos+S0.length(),  S.length());  System.out.println(S1);  } | | package main  import (  "fmt"  "strings"  "os"  "bufio"  )  func main() {  reader := bufio.NewReader(os.Stdin)  fmt.Println("Enter S: ")  S, \_:= reader.ReadString('\n')  fmt.Println("Enter S0: ")  S1, \_ := reader.ReadString('\n')  var m,n, k int=0,0,0  n=strings.Index(S,S1)  m=n  for len(S)>1 && n!=-1{  m=n+ len(S1)  S=S[m:]  n=strings.Index(S,S1)  k++  }  if k<=1{  m=m-len(S1)  }  fmt.Println(m)  } | |

## **Анализ и преобразование слов в строке**

Во всех заданиях данного пункта предполагается, что исходные строки являются непустыми и не содержат начальных и конечных пробелов.

1. Дана строка, состоящая из русских слов, разделенных пробелами (одним или несколькими). Найти количество слов в строке.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  S: String;  i, Num: integer;  **begin**  Write('S: ');  Readln(S);  i := 1;  Num := 0;  **while** (i <= Length(S)) **do**  **begin**  **if** (S[i]=' ') **and** (S[i –  1] <> ' ') **then**  inc(Num);  inc(i);  **end**;  Writeln(Num + 1);  **end**.  Вариант 2.  **begin**  WriteLn('Введено слов: ', ReadLnString('Введите текст:').ToWords(' ').Count);  **end**. | | #include "stdafx.h"  #include <iostream>  #include <string>  #include <vector>  using namespace std;  const vector<string> Split(const string& s, const char& c)  {  string buff="";  vector<string> v;  for (auto n : s)  {  if (n != c)  buff += n;  else if (n == c && buff != "") {  v.push\_back(buff);  buff = "";  }  }  if (buff != "")  v.push\_back(buff);  return v;  }  int main(){  string S;  getline(cin,S);  vector<string> v=Split(S, ' ');  cout << v.size() << endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| S=input("S->")  x=S.split(" ")  k=0  for i in range(0,len(x)):  if x[i]!="":  k+=1  print(k) | static void Main(string[] args) {  string S =Console.ReadLine();  string []x=S.Split(new char[] { ' ' });  Console.WriteLine(x.Count());  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  String [] x=S.split(" ");  System.out.println(x.length);  } | | | package main  import (  "fmt"  "strings"  "os"  "bufio"  )  func main() {  reader := bufio.NewReader(os.Stdin)  fmt.Println("Enter S: ")  S,\_:=reader.ReadString('\n')  a := strings.Split(S, " ")  fmt.Println(len(a))  } |

1. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Найти количество слов, которые содержат хотя бы одну букву «А».

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Begin**  **var** S:=ReadLnString(S:');  **var** S1 := S.ToWords(' ');  **var** k := 0;  **for var** i := 0 **to** S1.Count - 1 **do**  **if** s1[i].Contains('А') **then**  k += 1;  Writeln(k);  **end**. | | #include <iostream>  #include <string>  #include <vector>  using namespace std;  const vector<string> Split(const string& s, const char& c)  {  string buff="";  vector<string> v;  for (auto n : s)  {  if (n != c)  buff += n;  else if (n == c && buff != "") {  v.push\_back(buff);  buff = "";  }  }  if (buff != "")  v.push\_back(buff);  return v;  }  int main(){  string S;  getline(cin,S);  int k = 0;  vector<string> v=Split(S, ' ');  for (auto ss:v) {  if ((ss.find("A") >= 0) &&  (ss.find("A") < ss.size()))  k++;  }  cout << k << endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| S=input("S->")  x=S.split(" ")  k=0  for i in range(0,len(x)):  if x[i].find("А")!=-1:  k+=1  print(k) | static void Main(string[] args)  {  string S =Console.ReadLine();  string []x=S.Split(new char[] { ' ' });  int k = 0;  for (int i=0; i<x.Count(); i++)  if (x[i].IndexOf('а') >= 0)  k++;  Console.WriteLine(k);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  String [] x=S.split(" ");  int k=0;  for (int i=0; i<x.length; i++)  if (x[i].indexOf('A')>=0)  k++;  System.out.println(k);  } | | | package main  import (  "fmt"  "strings"  "os"  "bufio"  )  func main() {  reader := bufio.NewReader(os.Stdin)  fmt.Println("Enter S: ")  S, \_ := reader.ReadString('\n')  a := strings.Split(S, " ")  var k int=0  for i:=0; i<len(a); i++{  if strings.Contains(a[i], "A"){  k++  }  }  fmt.Println(k)  } |

1. Дана строка, состоящая из русских слов, набранных заглавными буквами и разделенных пробелами (одним или несколькими). Преобразовать каждое слово в строке, заменив в нем все последующие вхождения его первой буквы на символ «.» (точка). Например, слово «МИНИМУМ» надо преобразовать в «МИНИ.У.». Количество пробелов между словами не изменять.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  S: String;  i: integer;  C1: Char;  **begin**  Write('S: ');  Readln(S);  C1 := S[1];  i := 2;  **while** (i <= Length(S)) **do**  **begin**  **if** (S[i - 1] = ' ') **and**  (S[i] <> ' ') **then**  C1 := S[i]  **else if** S[i] = C1 **then**  S[i] := '.';  inc(i);  **end**;  Writeln(S);  **end**. | | #include <iostream>  #include <string>  #include <vector>  using namespace std;  const vector<string> Split(const string& s, const char& c){  string buff = "";  vector<string> v;  for (auto n : s)  {  if (n != c)  buff += n;  else if (n == c && buff != "") {  v.push\_back(buff);  buff = "";  }  }  if (buff != "")  v.push\_back(buff);  return v;  }  int main()  {  string S;  getline(cin, S);  int k = 0;  vector<string> v = Split(S, ' ');  for (string ss : v) {  for (int i = 1; i < ss.length(); i++)  if (ss[i] == ss[0])  ss.replace(i, 1, ".");  cout << ss << endl;  }  system("pause");  return 0;  } | |
| Python | C# | | |
| S=input("S->")  x=S.split(" ")  for i in range (0, len(x)):  c=x[i][0]  ss=x[i][1:];  ss=c+ss.replace(c, ".")  print(ss) | static void Main(string[] args)  {  string S =Console.ReadLine();  string []x=S.Split(new char[] { ' ' });  for (int i=0; i<x.Count(); i++){  char c=x[i][0];  string Ss = c.ToString();  for (int j=1; j<x[i].Length; j++) {  if (x[i][j]==c)  Ss=Ss+".";  else  Ss=Ss+x[i][j];  }  x[i]=Ss;  }  S = "";  for (int j=0; j<x.Count(); j++) {  S=S+x[j]+" ";  Console.WriteLine(S);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args)  {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  String [] x=S.split(" ");  int k=0;  for (int i=0; i<x.length; i++)  {  char c=x[i].charAt(0);  String ss=""+x[i].charAt(0);  System.out.println(c);  for (int j=1; j<x[i].length(); j++)  if (x[i].charAt(j)==c)  ss=ss+".";  else  ss=ss+x[i].charAt(j);  x[i]=ss;  }  for (int j=0; j<x.length; j++)  System.out.println(x[j]);  } | | | package main  import (  "fmt"  "strings"  "os"  "bufio"  )  func main() {  reader := bufio.NewReader(os.Stdin)  fmt.Println("Enter S: ")  S, \_:=reader.ReadString('\n')  a := strings.Split(S, " ")  for i:=0; i<len(a); i++{  c:=string(a[i][0])  ss:=a[i][1:];  ss=c+strings.Replace(ss, c, ".", -1)  fmt.Println(ss)  }  } |

1. Дана строка-предложение на русском языке. Подсчитать количество содержащихся в строке знаков препинания.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Begin**  **var** S := ReadLnString('Введите текст:');  **var** k := 0;  **for var** i:=1 **to** S.Length **do**  **if** char.IsPunctuation(S[i]) **then**  k += 1;  Writeln(k);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main()  {  string S;  getline(cin, S);  int k = 0;  for (auto c: S) {  if (ispunct(c))  k++;  }  cout << k;  system("pause");  return 0;  } |
| Python | C# | |
| import string  S=input("S->")  k=0  for c in S:  if c in string.punctuation:  k+=1  print(k) | static void Main(string[] args)  {  string S =Console.ReadLine();  int i, k = 0;  for (i=0; i<S.Length; i++)  if (Char.IsPunctuation(S[i]))  k++;  Console.WriteLine(k);  Console.ReadKey();  } | |
| Java | | |
| package main  import (  "fmt"  "os"  "bufio"  "unicode"  )  func main()  {  reader := bufio.NewReader(os.Stdin)  fmt.Println("Enter S: ")  S, \_ := reader.ReadString('\n')  var k int =0  for i:=0; i<len(S); i++{  if unicode.IsPunct(rune(S[i])){  k++  }  }  fmt.Println(k)  } | | |

## **Дополнительные задания на обработку строк**

1. Дана строка, содержащая *полное имя файла*, то есть имя диска, список каталогов (путь), собственно имя и расширение. Выделить из этой строки имя файла (без расширения).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Begin**  **var** S := ReadLnString('Введите текст:');  **var** S1 := S.ToWords('\');  S := S1[S1.Count - 1];  **var** pos := pos('.', S);  delete(S,pos,S.Length-pos+1);  WriteLn(S);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main()  {  string S;  getline(cin, S);  int pos = S.find\_last\_of("/");  S = S.substr(pos + 1, S.size());  pos = S.find(".");  S = S.substr(0, pos);  cout << S;  system("pause");  return 0;  } | |
| Python | C# | | |
| import os  S=input("S->")  x=os.path.basename(S)  x=x[:x.find(".")]  print(x) | static void Main(string[] args)  {  string S =Console.ReadLine();  int pos=S.LastIndexOf('/');  string Ss=S.Substring(pos+1, S.Length-  pos-1);  Ss = Ss.Substring(0, Ss.IndexOf("."));  Console.WriteLine(Ss);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  int pos=S.lastIndexOf('/');  int pos1=S.lastIndexOf('.');  String Ss=S.substring(pos+1, pos1);  System.out.println(Ss);  } | | | package main  import (      "fmt"  "path/filepath"  "strings"  "bufio"  "os"  )  func main() {  reader := bufio.NewReader(os.Stdin)  fmt.Println("Path: ")  path, \_ := reader.ReadString('\n')  file := filepath.Base(path)  n:=strings.Index(file, ".")  file=file[0:n]  fmt.Println(file)  } |

1. Дана строка-предложение на русском языке. Зашифровать ее, выполнив циклическую замену каждой буквы на следующую за ней в алфавите и сохранив при этом регистр букв («А» перейдет в «Б», «а» — в «б», «Б» — в «В», «я» — в «а» и т. д.). Букву «ё» в алфавите не учитывать («е» должна переходить в «ж»). Знаки препинания и пробелы не изменять.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Begin**  **var** S:=ReadLnString('Введите  текст:');  **for var** i := 1 **to** S.Length **do**  **if** ord(S[i]) **in** [1040..1103] **then**  **begin**  **var** T := ord(S[i]);  T := T + 1;  **if** (T=1072) **or** (T=1104) **then**  T := T - 32;  S[i] := chr(T);  **end**;  WriteLn(S);  **end**. | | #include <iostream>  #include <string>  using namespace std;  int main()  {  string S;  getline(cin, S);  unsigned char c;  for (int i=0; i<S.length(); i++){  c = S[i];  int T = (int)c+1;  if (T == 160)  T = 128;  if (T == 240)  T = 160;  S[i] = (char)T;  }  cout << S << endl;  system("pause");  return 0;  } |
| Python | C# | |
| import os  S=input("S->")  S1=""  for i in range(0, len(S)):  t=ord(S[i])+1  if t==1104:  t=1072  elif t==1072:  t=1040  S1=S1+chr(t)  print(S1) | static void Main(string[] args) {  string S =Console.ReadLine();  string Ss = "";  for (int i=0; i<S.Length; i++){  int T=(int)S[i];  T+=1;  if (T == 1104)  T = 1072;  if (T==1072)  T=1040;  Ss=Ss+(char)T;  }  Console.WriteLine(Ss);  Console.ReadKey();  } | |
| Java | | |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  String Ss="";  for (int i=0; i<S.length(); i++){  int T = (int)S.charAt(i)+1;  System.out.println(T);  if (T == 1072)  T = 1040;  if (T == 1104)  T = 1072;  Ss=Ss+(char)T;  }  System.out.println(Ss);  } | | |

1. Дана строка, содержащая латинские буквы и скобки трех видов: «()», «[]», «{}». Если скобки расставлены правильно (то есть каждой открывающей соответствует закрывающая скобка того же вида), то вывести число 0. В противном случае вывести или номер позиции, в которой расположена первая ошибочная скобка, или, если закрывающих скобок не хватает, число –1.

| PascalABC.NET |
| --- |
| **Var**  i, Num1, Num2, Num3, error: integer;  **begin**  **var** S := ReadLnString('Введите текст:');  i := 1;  error := 0;  Num1 := 0;  Num2 := 0;  Num3 := 0;  **while** (i <= Length(S)) **do**  **begin**  **if** S[i] = '(' **then** Num1 := Num1 + 1;  **if** S[i] = ')' **then** Num1 := Num1 - 1;  **if** S[i] = '[' **then** Num2 := Num2 + 1;  **if** S[i] = ']' **then** Num2 := Num2 - 1;  **if** S[i] = '{' **then** Num3 := Num3 + 1;  **if** S[i] = '}' **then** Num3 := Num3 - 1;  **if** ((Num1<0) **or** (Num2<0) **or** (Num3<0)) **and** (error=0) **then**  error := i;  inc(i);  **end**;  **if** error <> 0 **then**  Writeln(error)  **else if** ((Num1 > 0) **or** (Num2 > 0) **or** (Num3 > 0)) **then**  Writeln(-1)  **else** Writeln(0);  **end**. |
| Python |
| import os  S=input("S->")  Num1, Num2, Num3,er,i=0,0,0,0,0  while i<len(S):  if S[i]=="(":  Num1+=1  if S[i]==")":  Num1-=1  if S[i]=="[":  Num2+=1  if S[i]=="]":  Num2-=1  if S[i]=="{":  Num3+=1  if S[i]=="}":  Num3-=1  if (Num1<0 or Num2<0 or Num3<0) and er==0:  er=i  i+=1  if er!=0:  print(er)  elif (Num1>0) or (Num2>0) or (Num3>0):  print(-1)  else:  print(0) |
| Go |
| package main  import (      "fmt"  "bufio"  "os"  "strings"  )  func main() {  reader := bufio.NewReader(os.Stdin)  fmt.Println("Enter S: ")  S, \_ := reader.ReadString('\n')  var Num1, Num2, Num3,er,i int=0,0,0,0,0  for i<len(S){  if strings.Compare(string(S[i]), "(")==0{  Num1++  }  if strings.Compare(string(S[i]), ")")==0{  Num1--  }  if strings.Compare(string(S[i]), "[")==0{  Num2++  }  if strings.Compare(string(S[i]), "]")==0{  Num2--  }  if strings.Compare(string(S[i]), "{")==0{  Num3++  }  if strings.Compare(string(S[i]), "}")==0{  Num3--  }  if (Num1<0 || Num2<0 || Num3<0) && er==0{  er=i  }  i++  }  if er!=0{  fmt.Println(er)  }else if (Num1>0 || Num2>0 || Num3>0) {  fmt.Println(-1)  } else{  fmt.Println(0)  }  } |

**Задачи для самостоятельной работы**

**Вариант 1.**

1. Подсчитать, сколько раз в данной строке встречается буква Ch, вводимая с клавиатуры.
2. Дана символьная строка, заканчивающаяся точкой. Подсчитать, сколько слов в строке.
3. Символьная строка содержит одно слово. Проверить, будет ли оно читаться одинаково справа налево и слева направо (т.е. является ли оно палиндромом).

**Вариант 2.**

1. Дан текст, в котором записано одно из стихотворений А.С. Пушкина. Сколько раз в каждой строке встречается данный символ?
2. Дана символьная строка, содержащая английский текст. Найти количество слов, начинающихся с буквы b.
3. В записке слова зашифрованы – каждое из них записано наоборот. Расшифровать сообщение.

**Вариант 3.**

1. Дан текст, в котором записано одно из стихотворений А.С. Пушкина. Сколько раз встречаются гласные буквы?
2. Дана символьная строка. Подсчитать в ней количество вхождений букв r, k, t.
3. Проверить, одинаковое ли число открывающихся и закрывающихся скобок в данной строке.

**Вариант 4**

1. Из заданной строки удалить среднюю букву, если длина строки нечетная, иначе — удалить две средние буквы.
2. Дана символьная строка. Определить, сколько в ней знаков «\*», «;», «:».
3. Символьная строка, содержащая произвольный русский текст, состоит не более, чем из 200 символов. Написать, какие буквы и сколько раз встречаются в этом тексте. Ответ должен приводиться в грамматически правильной форме, например: а – 25 раз, к – 3 раза.

**Вариант 5**

1. Заменить в заданной строке все буквы Ch1 на Ch2 (их значения вводить с клавиатуры).
2. Дана символьная строка, cодержащая текст, который заканчивается точкой. Найти длину самого короткого слова и самого длинного слова.
3. Упорядочить данный массив английских слов по алфавиту, рассматривая только первый символ каждого слова.

**Вариант 6**

1. Заменить все вхождения подстроки Str1 на подстроку Str2, которые вводятся с клавиатуры.
2. Дана строка символов, среди которых есть двоеточие (:). Определить, сколько символов ему предшествует.
3. Выписать из символьной строки слова, начинающиеся и заканчивающиеся на одну и ту же букву.

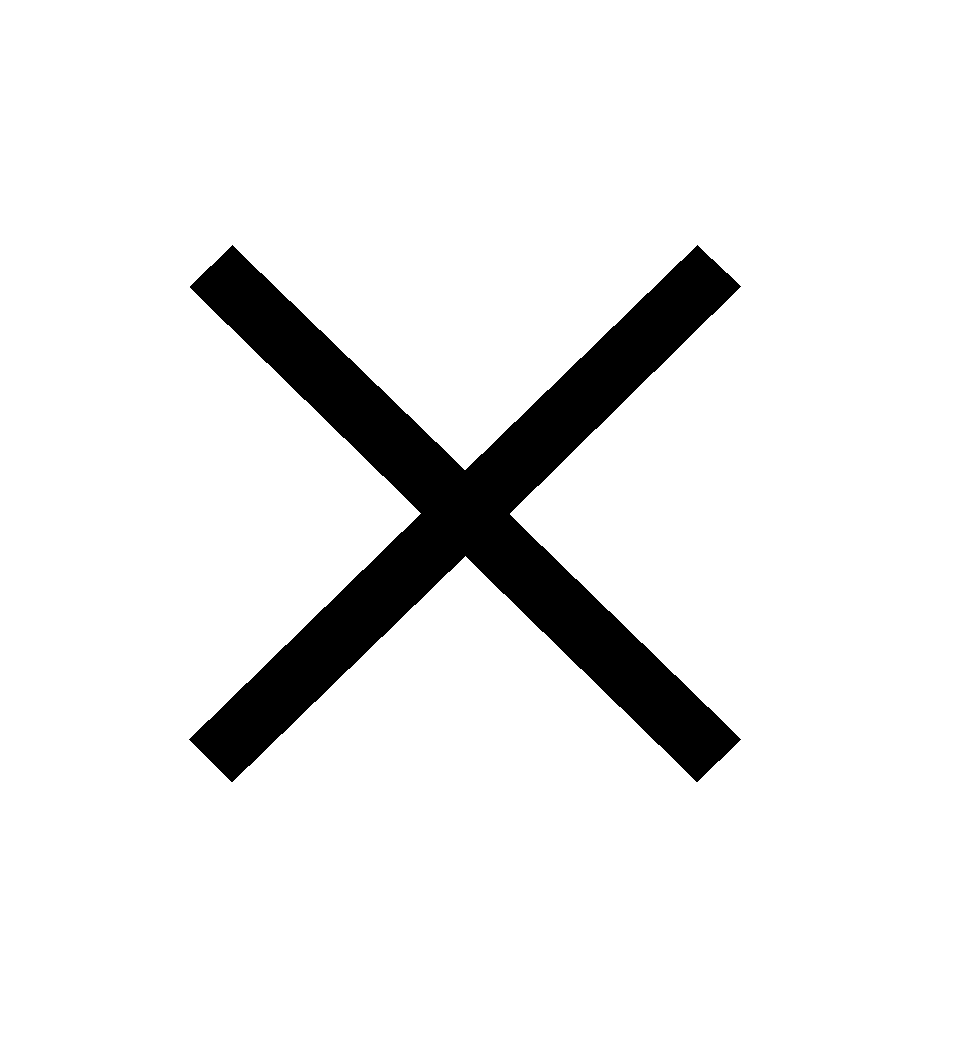
**Вариант 7**

1. В заданной строке после каждой буквы Ch вставить строку Str1.
2. Дана символьная строка, содержащая текст, заканчивающийся точкой. Вывести на экран слова, содержащие три буквы.
3. Даны две символьные строки А и В. Составьте программу, проверяющую, можно ли из букв, входящих в А, составить В. (буквы можно использовать не более одного раза и можно переставлять). Например, А = ИНТЕГРАЛ; В = АГЕНТ – составить можно; если В = ГРАФ – составить нельзя.

**Вариант 8**

1. В заданной строке удвоить каждое вхождение буквы Ch.
2. Дана символьная строка. Преобразовать ее, удалив каждый символ «\*» и повторив каждый символ, отличный от «\*».
3. Символьная строка содержит произвольный русский текст. Проверить, каких букв в нем больше: гласных или согласных.

**Вариант 9**

1. Даны две строки. Если они начинаются с одинаковых символов, то напечатать «ДА», иначе –«НЕТ».
2. Дана символьная строка. Определить, сколько раз входит в нее группа букв : abc.
3. Двумерный массив МN содержит некоторые буквы русского алфавита, расположенные в произвольном порядке. Написать программу, проверяющую можно ли из этих букв составить данное слово S.

**Вариант 10**

1. Дана последовательность слов. Напечатать все слова, отличные от слова «hello”.
2. Дана символьная строка. Подсчитать количество букв k в последнем ее слове.
3. Результаты вступительных экзаменов представлены в виде списка из N строк, в каждой строке которого записаны фамилия студента и отметки по каждому из М экзаменов. Определить количество абитуриентов, сдавших вступительные экзамены на «отлично».

**Вариант 11**

1. Дана последовательность слов. Напечатать все слова последовательности, которые встречаются в ней по одному разу.
2. Дана символьная строка. Подсчитать, сколько различных символов встречаются в ней. Вывести их на экран.
3. Составить программу преобразования натуральных чисел, записанных в римской нумерации, в десятичную систему счисления.

**Вариант 12**

1. Дано предложение. Напечатать все различные слова.
2. Дана строка символов. Подсчитать самую длинную последовательность подряд идущих букв а.
3. Из заданной символьной строки выбрать те символы, которые встречаются в ней только один раз, в том порядке, в котором они встречаются в тексте.

**Вариант 13**

1. Дана последовательность слов. Напечатать все слова, предварительно преобразовав каждое из них по следующему правилу: удалить из слова все предыдущие вхождения последней буквы.
2. Дана строка символов, среди которых есть одна открывающаяся и одна закрывающаяся скобка. Вывести на экран все символы, расположенные внутри этих скобок.
3. В символьном массиве хранятся фамилии и инициалы учеников класса. Требуется напечатать список класса с указанием для каждого ученика количества его однофамильцев.

**Вариант 14**

1. Дана последовательность слов. Напечатать все слова, предварительно преобразовав каждое из них по следующему правилу: оставить в слове только первые вхождения каждой буквы.
2. Имеется строка символов, содержащая буквы латинского алфавита и цифры. Вывести на экран длину наибольшей последовательности цифр, идущих подряд.
3. Дано число в двоичной системе счисления. Проверить правильность ввода этого числа (в его записи должны быть только символы 0 и 1). Если число введено неверно, повторить ввод. При правильном вводе перевести число в десятичную систему счисления.

**Вариант 15**

1. Дана последовательность слов. Напечатать те слова последовательности, которые отличны от последнего слова и удовлетворяют следующему свойству: в слове нет повторяющихся букв.
2. Дан набор слов, разделенных точкой с запятой (;). Набор заканчивается двоеточием (:). Определить, сколько в нем слов, заканчивающихся буквой а.
3. Программа. Напечатать заданный непустой текст, заменив в нем все пары ph на букву f.

**Вариант 16**

1. Дана последовательность слов. Напечатать те слова последовательности, которые отличны от последнего слова и удовлетворяют следующему свойству: буквы слова упорядочены по алфавиту.
2. Дана строка символов, заканчивающаяся точкой. Указать те слова, которые содержат хотя бы одну букву k.
3. Напечатать заданный текст, удалив из него лишние пробелы, т. е. из нескольких подряд идущих пробелов оставить только один.

**Вариант 17**

1. Дана последовательность слов. Напечатать те слова последовательности, которые отличны от последнего слова и удовлетворяют следующему свойству: слово совпадает с начальным отрезком латинского алфавита (a, ab, abc, abсd,...).
2. Символьная строка заканчивается восклицательным знаком (!). Слова в ней отделены друг от друга точкой с запятой (;). Найти те слова, которые начинаются и заканчиваются одной и той же буквой.
3. Заданный текст распечатать по строкам, понимая под строкой либо очередные 60 литер, если среди них нет запятой, либо часть текста до запятой включительно.

**Вариант 18**

1. Дана последовательность слов. Напечатать те слова последовательности, которые отличны от последнего слова и удовлетворяют следующему свойству: слово симметрично.
2. В символьной строке заменить все двоеточия (:) точкой с запятой (;). Подсчитать количество замен.
3. Программа. Дана непустая последовательность непустых слов из латинских букв; соседние слова отделены друг от друга запятой, за последним словом—точка. Определить количество слов, которые начинаются с буквы а;

**Вариант 19**

1. Составьте программу вывода самой большой цифры в записи заданного числа.
2. В заданной строке удалить все символы двоеточие (:) и подсчитать количество удаленных символов.
3. Программа. Дана непустая последовательность непустых слов из латинских букв; соседние слова отделены друг от друга запятой, за последним словом—точка. Определить количество слов, которые оканчиваются буквой w;

**Вариант 20**

1. Найти сумму всех чисел строки.
2. В символьной строке между словами вставить вместо пробела запятую и пробел.
3. Программа. Дана непустая последовательность непустых слов из латинских букв; соседние слова отделены друг от друга запятой, за последним словом—точка. Определить количество слов, которые начинаются и оканчиваются одной и той же буквой;

**Вариант 21**

1. Ввести предложение, слова в которых разделены пробелами и запятыми. Распечатать это предложение, удалив из него те слова, которые встретились там более одного раза.
2. Удалить часть символьной строки, заключенную в скобки (вместе со скобками).
3. Программа. Дана непустая последовательность непустых слов из латинских букв; соседние слова отделены друг от друга запятой, за последним словом—точка. Определить количество слов, которые содержат хотя бы одну букву d;

**Вариант 22**

1. Даны две символьные строки, состоящие только из цифр (длина каждой более 10 символов). Считая, что в этих строках находятся очень длинные числа, сформировать третью строку- сумму этих чисел.
2. Определить, сколько раз в строке встречается заданное слово.
3. Программа. Дана непустая последовательность непустых слов из латинских букв; соседние слова отделены друг от друга запятой, за последним словом—точка. Определить количество слов, которые содержат ровно три буквы е.

**Вариант 23**

1. Дан произвольный текст. Отредактировать текст так, чтобы:

* между словами был ровно один пробел;
* предложения в тексте разделялись ровно двумя пробелами.

1. В заданной строке имеется одна точка с запятой. Подсчитать количество символов до точки с запятой и после нее.
2. Значениями литерных переменных с2, с1 и с0 являются цифры. Присвоить целой переменной k число, составленное из этих цифр (например, если с2=='8', с1='0' с0='5', то k=805).

**Вариант 24**

1. Ввести два предложения и распечатать самые длинные слова, общие для этих предложений. Если нужных слов нет - сообщить об этом.
2. Дана строка символов. Преобразовать ее, заменив все двоеточия (:), встречающиеся среди первых n/2 символов, на точку с запятой (;), и заменив точками все восклицательные знаки, встречающиеся среди символов, стоящих после n/2 символов.
3. Присвоить литерным переменным с2, с1 и с0 соответственно левую, среднюю и правую цифры трехзначного числа k.

**Вариант 25**

1. Дана последовательность слов. Напечатать все слова в алфавитном порядке.
2. Вычислить s—сумму порядковых номеров всех букв, входящих в слово SUM.
3. Используя только литерный ввод, т. е. процедуру read(c), где с—литерная переменная, ввести непустую последовательность цифр, перед которой может находиться знак “+” или “—” и за которой следует пробел, и, получив соответствующее целое число, присвоить его целой временной k.

# **Двоичные (типизированные) файлы: группа File**

Условие вида «дан файл (целых чисел, вещественных чисел и т. д.)» означает, что в наборе исходных данных указано *имя* данного файла (текстовая строка), а сам исходный файл существует и находится в текущем каталоге. Если существование исходных файлов требуется проверять в ходе выполнения задания, то это особо оговаривается в формулировке задания.

Если в задании требуется создать новый файл, то имя создаваемого файла также входит в набор исходных данных и обычно является *последним* элементом этого набора.

Под размером двоичного типизированного файла всегда подразумевается количество содержащихся в нем *элементов* указанного типа (а не количество байтов, как это принято в операционной системе). В заданиях предполагается, что файловые элементы нумеруются от 1.

Если о минимальном размере исходного файла в задании ничего не сказано, то предполагается, что он равен 2 (то есть файл содержит по крайней мере два элемента). Максимальный размер исходных файлов не устанавливается, поэтому при решении заданий не следует использовать вспомогательные массивы, содержащие все элементы исходных файлов, однако допускается использование *вспомогательных файлов*.

## **Основные операции с двоичными файлами**

### **Создание файла, ввод и вывод его элементов**

1. Дана строка *S*. Если *S* является допустимым именем файла, то создать пустой файл с этим именем и вывести True. Если файл с именем *S* создать нельзя, то вывести False.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  S: String;  F: **file**;  **begin**  Write('S: ');  Readln(S);  **try**  Assign(F, S);  ReWrite(F);  Writeln(TRUE);  Close(F);  **except**  **on** e: Exception **do**  writeln(FALSE);  **end**  **end**. | | #include <iostream>  #include <string>  #include <fstream>  using namespace std;  int main(){  fstream file;  string S;  getline(cin, S);  file.open(S, ios\_base::out |  ios\_base::in);  if (file.is\_open())  cout << "Yes"<<endl;  else cout << "No"<<endl;  system("pause");  } | |
| Python | C# | | |
| from pathlib import Path  my\_file = Path(input("FileName:"))  print(my\_file.is\_file()) | static void Main(string[] args){  string myFile =Console.ReadLine();  if(File.Exists(@myFile))  Console.WriteLine("Yes");  else Console.WriteLine("No");  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  Scanner sc=new Scanner(System.in);  String filename=sc.nextLine();  File file=new File(filename);  if (file.exists())  System.out.println("Yes");  else  System.out.println("No");  } | | | package main  import (  "fmt"  "os"  )  func main() {  file, err := os.Create("D://hello.txt") // создаем файл  if err != nil{ // если возникла ошибка  fmt.Println("Yes:", err)  os.Exit(1) // выходим из программы  }  defer file.Close() // закрываем файл  fmt.Println("No") // hello.txt  } |
| Kotlin | | | |
| **import** java.util.Scanner **import** java.io.File **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = reader.next()  **var** fileObject = File(fileName)  *// create a new file* **val** isNewFileCreated :Boolean = fileObject.createNewFile()  **if**(isNewFileCreated){  *print*(**"Yes"**)  } **else** {  *print*(**"No"**)  } } | | | |

1. Дано имя файла и вещественные числа *A* и *D*. Создать файл вещественных чисел с данным именем и записать в него 10 первых членов *арифметической прогрессии* с начальным членом *A* и разностью *D*:

*A*,    *A* + *D*,    *A* + 2·*D*,    *A* + 3·*D*,    … .

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  S: String;  F: **file of** Real;  i: integer;  A, D, Num: Real;  **begin**  Write('S: ');  Readln(S);  Write('A,D: ');  Readln(A, D);  Assign(F, S);  ReWrite(F);  **for** i := 0 **to** 9 **do**  **begin**  Num := A + i \* D;  Write(F, Num);  **end**;  Close(F);  **end**. | | #include <iostream>  #include <string>  #include <fstream>  using namespace std;  int main()  {  string S;  float a, d;  getline(cin, S);  ofstream of(S);  if(of.is\_open())  {  cin >> a >> d;  for (int i=0; i<10; i++)  of << a+i\*d<<" ";  of.close();  }  else  cout << "Unable to open file";  system("pause");  } | |
| Python | C# | | |
| import pickle  FILENAME = "user.dat"  a=float(input("->"))  d=float(input("->"))  with open(FILENAME, "wb")  as file:  for i in range(0,10):  pickle.dump(a+i\*d, file) | static void Main(string[] args)  {  string fileName = @"D:/out.txt";  BinaryWriter writer = new BinaryWriter(File.Open(fileName, FileMode.Create));  double d, a;  a=Convert.ToDouble(Console.ReadLine());  d=Convert.ToDouble(Console.ReadLine());  for (int i=0; i<10; i++)  writer.Write(a + i \* d);  writer.Close();  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) throws FileNotFoundException, IOException {  double a,d;  Scanner sc=new Scanner(System.in);  a=sc.nextDouble();  d=sc.nextDouble();  String filename = "D:\\input.txt";  RandomAccessFile file=new  RandomAccessFile(new  File(filename),"rw");  for (int i=0; i<10; i++)  file.writeDouble(a+i\*d);  } | | | package main  import (  "fmt"  "os"  )  func main() {  var a,d float64  fmt.Scan(&a,&d)  file, err := os.Create("person.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  defer file.Close()  for i:=0; i<10;i++{  f := float64(i)  fmt.Fprintf( file,  "%f", a+f\*d)  }  } |
| Kotlin | | | |
| **import** java.util.Scanner **import** java.io.File **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = reader.next()  **val** myfile = File(fileName)  **var** a=reader.nextDouble()  **var** d=reader.nextDouble()  myfile.*printWriter*().*use* **{** out **->  for** (i **in** (1..10))  out.print((a+i\*d).toString()+**" "**)  **}** *println*(**"Writed to file"**) } | | | |
| VB.NET | | | |
| Sub Main()  Dim a, d As Single  Dim writer As BinaryWriter  writer = New BinaryWriter(File.Open("D:/output.dat",  FileMode.Create))  a = Convert.ToSingle(Console.ReadLine())  d = Convert.ToSingle(Console.ReadLine())  For i As Integer = 0 To 10  writer.Write(a + i \* d)  Next  writer.Close()  Console.ReadKey()  End Sub | | | |

1. Дано целое число *K* и файл, содержащий неотрицательные целые числа. Вывести *K*-й элемент файла (элементы нумеруются от 1). Если такой элемент отсутствует, то вывести –1.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  F: **file of** Integer;  i, Num: integer;  **begin**  **var** S := ReadLnString('Введите имя файла:');  **var** K := ReadLnInteger(K:');  **if** fileexists(S) **then**  **begin**  Assign(F, S);  Reset(F);  Num := 1;  **while** (**not** eof(F)) **and**  (Num <= K) **do**  **begin**  Read(F, i);  inc(Num);  **end**;  **if** Num - 1 <> K **then**  Writeln(-1)  **else** writeln(i);  Close(F);  **end**;  **end**. | | #include <iostream>  #include <string>  #include <fstream>  using namespace std;  int main()  {  string S;  int k,a;  getline(cin, S);  ifstream ifs(S);  if(ifs.is\_open())  {  cin >> k;  k = k == 2 ? 1 :  (k - 2) \* sizeof(int);  ifs.seekg(k, ios::beg);  ifs >> a;  cout << a<<" ";  }  else  cout << "Unable to open file";  system("pause");  } | |
| Python | C# | | |
| import pickle  FILENAME = "user.dat"  k=int(input("k->"))  i=0  with open(FILENAME,"rb")  as f:  while i<k:  x=int(pickle.load(f))  i+=1  print(x) | static void Main(string[] args){  string fileName = @"D:/out.txt";  BinaryReader reader = new  BinaryReader(File.Open(fileName,  FileMode.Open));  int k;  k=Convert.ToInt32(Console.ReadLine());  reader.BaseStream.Seek(k\*sizeof(int),  SeekOrigin.Begin);  int a = reader.ReadInt32();  Console.WriteLine(a);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) throws FileNotFoundException, IOException {  int a, k, i=1;  Scanner sc=new Scanner(System.in);  k=sc.nextInt();  String filename = "D:\\input.txt";  RandomAccessFile file=new  RandomAccessFile(new  File(filename),"rw");  while (((a=file.readInt())!=-  1.0)&& (i<k))  i++;  System.out.println(a);  }  }  Варинат 2.  public static void main(String[] args) {  int a,k;  Scanner sc=new Scanner(System.in);  k=sc.nextInt();  String filename = "D:\\input.txt";  RandomAccessFile file=new  RandomAccessFile(new  File(filename),"rw");  file.seek((k-1)\*Integer.SIZE/8);  a=file.readInt();  System.out.println(a);  }  } | | | package main  import (  "fmt"  "os"  )  func main() {  file, \_ := os.Open("input.dat")  var x,k int64  fmt.Scan(&k)  defer file.Close()  file.Seek(k\*2,0);  fmt.Fscanf(file, "%v", &x)  fmt.Printf("%v ",x)  } |
| Kotlin | | | |
| **import** java.util.Scanner **import** java.io.File **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = **"D:/input.txt"**; *//reader.next()* **val** myfile = File(fileName)  **val** k = reader.nextInt()  **var** i = 0  **var** a=0;  **var** readFile = Scanner(myfile);  **while** ((readFile.hasNext()) && (i < k)) {  i++;  a = readFile.nextInt();  }  *println*(a); } | | | |
| VB.NET | | | |
| Sub Main()  Dim reader As BinaryReader  reader = New BinaryReader(File.Open("D:/output.dat",  FileMode.Open))  Dim k As Integer = Convert.ToInt32(Console.ReadLine())  reader.BaseStream.Seek((k - 1) \* 4, SeekOrigin.Begin)  Dim a As Integer = reader.ReadInt32()  Console.WriteLine(a)  reader.Close()  Console.ReadKey()  End Sub | | | |

1. Дан файл целых чисел. Создать новый файл, содержащий те же элементы, что и исходный файл, но в обратном порядке.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  S: String;  F\_in, F\_out: **file of**  Integer;  a: **array**[1..1000] **of** real;  i,j,R: integer;  **begin**  i := 1;  Write('File1: ');  Readln(S);  Assign(F\_in, S);  Reset(F\_in);  Write('File2: ');  Readln(S);  Assign(F\_out, S);  ReWrite(F\_out);  **while not** Eof(F\_in) **do**  **begin**  Read(F\_in, R);  a[i] := R;  i += 1;  **end**;  **for** j:=i-1 **downto** 1 **do**  Write(F\_out, a[j]);  Close(F\_in);  Close(F\_out);  **end**. | #include <iostream>  #include <string>  #include <fstream>  #include<vector>  using namespace std;  int main()  {  string S;  int i,a;  getline(cin, S);  ifstream ifs(S);  ofstream ofs("D:/output.txt");  vector<int>v;  if(ifs.is\_open())  {  while (!ifs.eof()) {  ifs >> a;  v.push\_back(a);  }  }  else  cout << "Unable to open file";  for (i = v.size() - 1; i >= 0; i--)  ofs << v[i] << " ";  ofs.close();  system("pause");  } |
| Python | |
| import pickle  x = []  with open("D:/input.txt") as f:  for line in f:  x.append([int(x) for x in line.split()])  with open("D:/output.txt", 'w') as f2:  for i in range(len(x[0])-1, 0,-1):  f2.write(str(x[0][i])+" ") | |
| Go | |
| package main  import (  "fmt"  "os"  "io"  )  func main() {  file, err := os.Open("input.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  file1, er1 := os.Create("output.dat")  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  defer file.Close()  defer file1.Close()  a:=[100]int{0};  var i int=0  for{  \_, err = fmt.Fscanf(file, "%d", &a[i])  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  i++  }  for j:=i-1; j>=0; j--{  fmt.Fprintf( file,"%d ", a[j])  }  } | |

1. Дан файл целых чисел. Найти количество содержащихся в нем *серий* (то есть наборов последовательно расположенных одинаковых элементов). Например, для файла с элементами 1, 5, 5, 5, 4, 4, 5 результат равен 4.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Var**  S: String;  F\_in: **file of** Integer;  El, i, temp: integer;  **begin**  Assign(F\_in, 'D:\INPUT.txt');  Reset(F\_in);  Read(F\_in, temp);  El := temp;  i := 1;  **while** (**not** eof(F\_in)) **do**  **begin**  Read(F\_in, temp);  **if** El <> temp **then**  **begin**  El := temp;  inc(i);  **end**;  **end**;  Writeln(i);  Close(F\_in);  **end**. | | #include <fstream>  #include<vector>  using namespace std;  int main()  {  string S;  int a, b, k=1;  getline(cin, S);  ifstream ifs(S);  ifs >> b;  while (!ifs.eof()) {  ifs >> a;  if (b!= a) {  b = a;  k++;  }  }  cout << k << " \n ";  system("pause");  } |
| Python | | |
| import pickle  FILENAME = "input.dat"  a=[]  with open(FILENAME, "rb") as file:  for i in range(0,10):  x=pickle.load(file)  a.append(x);  print(a);  a.reverse();  print(a);  with open(FILENAME, "wb") as file:  for i in range(0,10):  pickle.dump(a[i]) | | |
| Java | Go | |
| public static void main(String[] args) throws FileNotFoundException, IOException {  int a,b, k=0;  Scanner sc=new Scanner(System.in);  File filename =new File("D:\\input.txt");  Scanner file=new Scanner(filename);  b=file.nextInt();  while (file.hasNext()) {  a=file.nextInt();  if (b!= a) {  b = a;  k++;  }  }  System.out.println(k);  } | package main  import (  "fmt"  "os"  "io"  )  func main() {  filein, err := os.Open("input.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  fileout, er1 := os.Create("output.dat")  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  defer filein.Close()  defer fileout.Close()  var a,b int64  var k int=1  fmt.Fscanf(filein, "%d",&a)  b=a  for{  \_, err = fmt.Fscanf(filein, "%d",&a)  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  if b!=a{  b=a  k++  }  }  fmt.Fprintf(fileout,"%d ",k)  } | |

1. Дан файл вещественных чисел. Создать файл целых чисел, содержащий длины всех убывающих последовательностей элементов исходного файла. Например, для исходного файла с элементами 1.7, 4.5, 3.4, 2.2, 8.5, 1.2 содержимое результирующего файла должно быть следующим: 3, 2.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  F\_in: **file of** Real;  F\_out: **file of** Integer;  El1, El: Real;  len: integer;  **begin**  Assign(F\_in, 'D:\input.txt');  Reset(F\_in);  Assign(F\_out, 'D:\output.txt');  ReWrite(F\_out);  Read(F\_in, El);  len := 1;  **while** (**not** eof(F\_in)) **do**  **begin**  El1 := El;  Read(F\_in, El);  **if** El1 > El **then**  inc(len)  **else**  **begin**  **if** len > 1 **then**  **begin**  Write(F\_out, len);  len := 1;  **end**;  **end**;  **end**;  Close(F\_in);  Close(F\_out);  **end**. | | #include <iostream>  #include <string>  #include <fstream>  using namespace std;  int main(){  string S, S1;  float a, b, k=1;  getline(cin, S);  ifstream ifs(S);  getline(cin, S1);  ofstream ofs(S1);  ifs >> a;  while (!ifs.eof()) {  b = a;  ifs >> a;  if (b > a) {  k++;  }  else  if (k > 1) {  ofs << k << " ";  k = 1;  }  }  system("pause");  } | |
| Python | C# | | |
| FILENAME = "D:/input.txt"  with open(FILENAME) as f:  for line in f:  x=([float(x) for x in line.split()])  m=len(x)  len=1  a=x[0]  for i in range (1, m-1):  b=a  a=x[i]  if b>a:  len+=1  else:  if len>1:  with open("D:/output.txt", 'a') as f2:  f2.write(str(len)+" ")  len=1 | static void Main(string[] args) {  string fileName = @"D:/out.txt";  string fileName1 = @"D:/out1.txt";  BinaryReader reader = new BinaryReader(File.Open(fileName, FileMode.Open));  BinaryWriter writer = new BinaryWriter(File.Open(fileName1, FileMode.Create));  double a=reader.ReadDouble();  double b;  int k=1,pos=sizeof(double);  int length = (int)reader.BaseStream.Length;  while (pos < length)  {  b = a;  a=reader.ReadDouble();  if (b > a)  {  k++;  }  else  if (k > 1) {  writer.Write(k);  k = 1;  }  pos += sizeof(double);  }  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) {  double a,b;  int k=1;  File filename =new File  ("D:\\input.txt");  Scanner file=new Scanner(filename);  DataOutputStream dos = new  DataOutputStream(new FileOutputStream  ("D:\\out1.txt"));  a=file.nextDouble();  while (file.hasNextDouble()) {  b=a;  a=file.nextDouble();  if (b > a) {  k++;  }  else  if (k > 1) {  dos.write(k);  k = 1;  }  }  } | | | package main  import (  "fmt"  "os"  "io"  )  func main() {  filein, err := os.Open("input.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  fileout, er1 := os.Create("output.dat")  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  defer filein.Close()  defer fileout.Close()  var a,b float64  var k int=1  fmt.Fscanf(filein, "%f",&a)  for{  b=a  \_, err = fmt.Fscanf(filein, "%f",&a)  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  if b>a{  k++  }else{  if k>1{  fmt.Fprintf(fileout,"%d", k)  fmt.Println(k)  k=1  }  }  }  } |

### **Преобразование файла**

1. Дан файл вещественных чисел. Заменить в нем все элементы на их квадраты.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  F\_in, F\_temp: **file of** Real;  temp: Real;  **begin**  Assign(F\_in, 'D:\input.txt');  Reset(F\_in);  Assign(F\_temp, 'D:\input1.txt');  ReWrite(F\_temp);  **while** (**not** eof(F\_in)) **do**  **begin**  Read(F\_in, temp);  temp := sqr(temp);  Write(F\_temp, temp);  **end**;  Close(F\_in);  Close(F\_temp);  Erase(F\_in);  Rename(F\_temp, 'D:\input.txt');  **end**. | #include <iostream>  #include <string>  #include <fstream>  #include<vector>  using namespace std;  int main()  {  string S;  int i, a;  getline(cin, S);  ifstream ifs(S);  vector<int>v;  while (!ifs.eof()) {  ifs >> a;  v.push\_back(a);  }  ifs.close();  ofstream ofs(S);  for (i = 0; i<v.size(); i++)  ofs << v[i]\*v[i]<< " ";  ofs.close();  system("pause");  } |
| Python | C# |
| import pickle  FILENAME = "user.dat"  f = open(FILENAME, "r+b")  s = f.read(8)  while s:  x = pickle.unpack("d", s)  f.seek(-8,1)  f.write(pickle.pack("d",x[0]\*\*2))  # Метод flush () очищает внутренний буфер  f.flush()  s = f.read(8)  f.close() | static void Main(string[] args)  {  string fileName = @"D:/out.txt";  string fileName1 = @"D:/out1.txt";  BinaryReader reader = new BinaryReader(File.Open(fileName, FileMode.Open));  BinaryWriter writer = new BinaryWriter(File.Open(fileName1, FileMode.Create));  int pos = 0;  double a;  int length = (int)reader.BaseStream.Length;  while (pos < length){  a=reader.ReadDouble();  writer.Write(a\*a);  pos += sizeof(double);  }  reader.Close();  writer.Close();  File.Delete(fileName);  File.Move(fileName1, fileName);  Console.ReadKey();  } |
| Java | |
| package main  import (  "fmt"  "os"  "io"  )  func main() {  filein, err := os.Open("input.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  fileout, er1 := os.Create("output.dat")  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  var a float64  for{  \_, err = fmt.Fscanf(filein, "%f",&a)  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  fmt.Fprintf(fileout, "%f ", a\*a)  }  filein.Close()  fileout.Close()  os.Remove("input.dat");  os.Rename("output.dat","input.dat")  os.Remove("output.dat");  } | |

1. Дан файл целых чисел. Удалить из него все элементы с четными номерами.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Var**  F\_in, F\_out: **file of** Integer;  El, i, len: integer;  **begin**  Assign(F\_in, 'D:\input.txt');  Reset(F\_in);  Assign(F\_out, 'D:\input1.txt');  ReWrite(F\_out);  i := 1;  **while not** eof(F\_in) **do**  **begin**  Read(F\_in, El);  **if** odd(i) **then**  Write(F\_out, El);  inc(i);  **end**;  Close(F\_in);  Close(F\_out);  Erase(F\_in);  Rename(F\_out, 'D:\input.txt');  **end**. | | #include <iostream>  #include <string>  #include <fstream>  #include<vector>  using namespace std;  int main()  {  string S;  int i, a;  getline(cin, S);  ifstream ifs(S);  vector<int>v;  while (!ifs.eof()) {  ifs >> a;  v.push\_back(a);  }  ifs.close();  ofstream ofs(S);  for (i = 0; i < v.size(); i++)  if (i % 2 == 0)  ofs << v[i] << " ";  ofs.close();  system("pause");  } |
| Python | | |
| import pickle  FILENAME = "user.dat"  f = open(FILENAME, "r+b")  s = f.read(8)  i=0  while s:  x.append(pickle.unpack("d", s))  s = f.read(8)  print(x)  for j in range(0,len(x)-1):  if j%2==1:  f.write(pickle.pack("d", x[j][0]\*\*2))  f.flush()  f.close() | | |
| Java | Go | |
| public static void main(String[] args) **throws FileNotFoundException, IOException** {  File file1 =new File("D:\\input.txt");  File file2 =new File("D:\\output.txt");  FileWriter fw=new FileWriter(file2);  int k=0, a;  Scanner fi=new Scanner(file1);  BufferedWriter writer = new BufferedWriter(fw);  while (fi.hasNextInt()) {  k++;  a=fi.nextInt();  if (k%2==0) writer.write(String.valueOf(a)+"");  }  writer.close();  fi.close();  file1.delete();  file2.renameTo(file1);  } | package main  import (  "fmt"  "os"  "io"  )  func main() {  filein, err := os.Open("input.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  var i int=0  fileout, er1 := os.Create("output.dat")  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  var a int64  for{  \_, err=fmt.Fscanf(filein, "%d",&a)  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  i++  if i%2==1{  fmt.Fprintf(fileout, "%d ", a)  }  }  filein.Close()  fileout.Close()  os.Remove("input.dat"); os.Rename("output.dat","input.dat")  os.Remove("output.dat");  } | |

### **Работа с нетипизированными двоичными файлами**

1. Даны два файла произвольного типа. Поменять местами их содержимое.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  Name1, Name2: String;  F\_1, F\_2: **file**;  **begin**  Name1 := 'D:\input.txt';  Name2 := 'D:\input1.txt';  Assign(F\_1, Name1);  Assign(F\_2, Name2);  F\_1.Rename('D:\temp.txt');  F\_2.Rename(Name1);  F\_1.Rename(Name2);  **end**. | | #include <iostream>  #include <cstdio>  using namespace std;  int main(){  string S= "D:/input.txt";  string S1 = "D:/input1.txt";  string temp = "D:/temp.txt";  rename(S.c\_str(), temp.c\_str());  rename(S1.c\_str(), S.c\_str());  rename(temp.c\_str(), S1.c\_str());  system("pause");  return 0;  } | |
| Python | C# | | |
| import os  FILENAME = "D:/input.txt"  FILENAME1 = "D:/input1.txt"  os.rename(FILENAME, "D:/temp.txt")  os.rename(FILENAME1,FILENAME)  os.rename("D:/temp.txt", FILENAME1) | static void Main(string[] args)  {  string fileName = @"D:/out.txt";  string fileName1 = @"D:/out1.txt";  string tmp = @"D:/temp.txt";  File.Move(fileName1, tmp);  File.Move(fileName, fileName1);  File.Move(tmp,fileName);  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) throws FileNotFoundException, IOException {  File file1 =new File("D:\\  input.txt");  File file2 =new File("D:\\  input1.txt");  File temp =new File("D:\\  temp.txt");  file1.renameTo(temp);  file2.renameTo(new File("D:\\  input.txt"));  temp.renameTo(new File("D:\\  input1.txt"));  } | | | package main  import "os"  func main() {  os.Rename("input.dat","temp.dat")  os.Rename("input1.dat","input.dat")  os.Rename("temp.dat","input1.dat")  os.Remove("temp.dat")  } |

**Замечание**. Записать/считать данные в в бинарном файле на С++.

| 1. Записать   #include <iostream>  #include <fstream>  using namespace std;  int main(){  ofstream ofs;  int x = 17;  ofs.open("junk.dat", ios::binary | ios::out);  for (int i = 2; i <= 20; i += 2) {  char x = i;  ofs.write(&x, sizeof(int));  }  ofs.close();  system("pause");  return 0;  }   1. Считать   #include <iostream>  #include <fstream>  using namespace std;  int main()  {  char x;  ifstream ifs;  ifs.open("junk.dat", ios::binary | ios::in);  while (!ifs.eof()) {  ifs.read(&x, sizeof(int));  cout << int(x)<<" ";  }  ifs.close();  system("pause");  return 0;  } |
| --- |

## **Обработка нескольких числовых файлов. Файлы-архивы**

1. Даны три файла целых чисел одинакового размера с именами *SA*, *SB*, *SC* и строка *SD*. Создать новый файл с именем *SD*, в котором чередовались бы элементы исходных файлов с одним и тем же номером:

*A*1,    *B*1,    *C*1,    *A*2,    *B*2,    *C*2,    … .

| PascalABC.NET |
| --- |
| **Var**  SA, SB, SC, SD: String;  FA, FB, FC, FD: **file of** Integer;  El: Integer;  **begin**  Write('SA: ');  Readln(SA);  Write('SB: ');  Readln(SB);  Write('SC: ');  Readln(SC);  Write('SD: ');  Readln(SD);  Assign(FA, SA);  Reset(FA);  Assign(FB, SB);  Reset(FB);  Assign(FC, SC);  Reset(FC);  Assign(FD, SD);  Rewrite(FD);  **while not** Eof(FA) **do**  **begin**  Read(FA, El);  Write(FD, El);  Read(FB, El);  Write(FD, El);  Read(FC, El);  Write(FD, El);  **end**;  Close(FA);  Close(FB);  Close(FC);  Close(FD);  **end**. |
| Python |
| import pickle  f1 = open("in1.dat", "rb")  f2 = open("in2.dat", "rb")  f3 = open("in3.dat", "rb")  f = open("out.dat", "w+b")  a = f1.read(8)  b= f2.read(8)  c= f3.read(8)  x,y,z=[],[],[]  while a and b and c:  x.append(pickle.unpack("d", a))  a = f1.read(8)  y.append(pickle.unpack("d", b))  b = f2.read(8)  z.append(pickle.unpack("d", c))  c = f3.read(8)  for i in range(0,len(x)-1):  f.write(pickle.pack('d', x[i][0]))  f.write(pickle.pack('d', y[i][0]))  f.write(pickle.pack('d', z[i][0])) |
| Go |
| package main  import (  "fmt"  "os"  "io"  )  func main() {  filein1, err := os.Open("input.dat")  filein2, \_ := os.Open("input1.dat")  filein3, \_ := os.Open("input2.dat")  fileout, er1 := os.Create("output.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  defer filein1.Close()  defer filein2.Close()  defer filein2.Close()  defer fileout.Close()  var a int64  for{  \_, err = fmt.Fscanf(filein1, "%d",&a)  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  fmt.Fprintf(fileout,"%d ",a)  fmt.Fscanf(filein2, "%d",&a)  fmt.Fprintf(fileout,"%d ",a)  fmt.Fscanf(filein3, "%d",&a)  fmt.Fprintf(fileout,"%d ",a)  }  } |

## **Cимвольные и строковые файлы**

*Строковым файлом* называется двоичный типизированный файл, элементами которого являются текстовые строки, хранящиеся в участках памяти *одинакового размера*.

1. Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные после первого символа пробела, включая и этот пробел.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  S: String;  F\_in, F\_out: **file of** Char;  C: Char;  **begin**  Write('File: ');  Readln(S);  Assign(F\_in, S);  Reset(F\_in);  Assign(F\_out, 'temp.txt');  ReWrite(F\_out);  Read(F\_in, C);  **while** (**not** eof(F\_in))  **and** (C <> ' ') **do**  **begin**  Write(F\_out, C);  Read(F\_in, C);  **end**;  Close(F\_out);  Close(F\_in);  Erase(F\_in);  Rename(F\_out, S);  **end**. | #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main(){  string S = "D:/input.txt";  ifstream file(S);  string str;  while ((file) && (str.length()==0))  file>>str;  file.close();  cout << str << "\n";  ofstream ofile(S,ios::out);  ofile << str;  ofile.close();  system("pause");  return 0;  } |
| Python | |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  f = open(f.name,"w")  for e in s:  f.write(e[:e.find(" ")]+"\n")  f.close() | |
| Java | |
| public static void main(String[] args) **throws FileNotFoundException, IOException** {  File file1 =new File("D:\\input.txt");  File file2 =new File("D:\\output.txt");  FileWriter fo=new FileWriter(file2);  Scanner fi=new Scanner(file1);  BufferedWriter writer = new BufferedWriter(fo);  String Ss=fi.next();  System.out.println(Ss);  writer.write(Ss);  writer.close();  file1.delete();  file2.renameTo(file1);  } | |
| Go | |
| package main  import (  "fmt"  "os"  "io"  )  func main() {  filein1, err := os.Open("input.dat")  fileout, er1 := os.Create("output.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  defer filein1.Close()  defer fileout.Close()  var a rune=5  for a!=32{  \_, err = fmt.Fscanf(filein1, "%c",&a)  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  if a!=32{  fmt.Fprintf(fileout,"%c",a)  }  }  os.Rename("output.txt", "input.txt")  os.Remove("output.txt");  } | |

1. Дан строковый файл, содержащий даты в формате «день/месяц/год», причем под день и месяц отводится по две позиции, а под год — четыре (например, «16/04/2001»). Создать два файла целых чисел, первый из которых содержит значения дней, а второй — значения месяцев для дат из исходного строкового файла (в том же порядке).

| PascalABC.NET | C++ |
| --- | --- |
| **Type**  st = String[100];  **var**  Name, S: St;  F\_in: **file of** St;  F\_out\_day, F\_out\_month:  **file of** integer;  temp, error: integer;  **begin**  Write('File In: ');  Readln(Name);  Assign(F\_in, Name);  Reset(F\_in);    Write('File Out Day: ');  Readln(Name);  Assign(F\_out\_day, Name);  Rewrite(F\_out\_day);  Write('File Out Month: ');  Readln(Name);  Assign(F\_out\_month, Name);  Rewrite(F\_out\_month);  **while** (**not** eof(F\_in)) **do**  **begin**  Read(F\_in, S);  Val(S[1] + S[2], temp,  error);  Write(F\_out\_day, temp);  Val(S[4] + S[5], temp,  error);  Write(F\_out\_month, temp);  **end**;  Close(F\_out\_month);  Close(F\_out\_day);  Close(F\_in);  **end**. | #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main()  {  string fileName = "D:/input.txt";  ifstream file(fileName);  ofstream ofile("D:/out1.txt",  ios::out);  ofstream ofile1("D:/out2.txt",  ios::out);  string str;  while (file) {  file >> str;  ofile<<str.substr(0,2)<<" ";  ofile1<<str.substr(3,2)<<" ";  }  file.close();  ofile.close();  ofile1.close();  system("pause");  return 0;  } |
| Python | |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  f1 = open("D:/out1.txt", "w")  f2 = open("D:/out2.txt", "w")  for e in s:  f1.write(e[0:2]+"\n")  f2.write(e[3:5]+"\n")  f1.close()  f2.close() | |
| Go | |
| package main  import (  "fmt"  "os"  "io"  "strings"  )  func main() {  filein1, err := os.Open("input1.dat")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  fileout, er1 := os.Create("output.dat")  fileout1, er2 := os.Create("output1.dat")  if er1 != nil{  fmt.Println(er1)  os.Exit(1)  }  if er2 != nil{  fmt.Println(er2)  os.Exit(1)  }  defer filein1.Close()  defer fileout.Close()  defer fileout1.Close()  var s string  for {  \_, err = fmt.Fscanln(filein1, &s)  if err != nil{  if err == io.EOF{  break  } else{  fmt.Println(err)  os.Exit(1)  }  }  fmt.Fprintf(fileout, "%v ",strings.Split(s, "/")[0])  fmt.Fprintf(fileout1,"%v ",strings.Split(s, "/")[1])  }  } | |

## **Использование файлов для работы с матрицами**

*Матрицей размера M×N* называется прямоугольная таблица чисел, содержащая *M* строк и *N* столбцов. Для работы с матрицами обычно используются *двумерные массивы*. Данный пункт посвящен способам обработки матриц, хранящихся в двоичных типизированных файлах на внешних носителях (дисках). Как и в остальных заданиях на обработку файловых данных, при выполнении заданий из данного пункта *не следует* использовать вспомогательные массивы, содержащие все файловые элементы.

В заданиях данного пункта используются дополнительные понятия теории матриц. Приведем определения этих понятий.

Пусть *A* — матрица размера M×N. Матрица *B* называется *транспонированной* к матрице *A*, если она имеет размер *N* ≠ *M* и ее элементы удовлетворяют следующему соотношению:

*BI*,*J* = *AJ*,*I*, *I* = 1, …, *N*, *J* = 1, …, *M*.

Пусть *A* — матрица размера M×N, *B* — матрица размера *N* ≠ *P*. Матрица *C* называется *произведением* матриц *A* и *B* (и обозначается *A*·*B*), если она имеет размер *M* ≠ *P* и ее элементы удовлетворяют следующему соотношению:

*CI,J* = *AI*,1 *B*1*,J* + *AI,*2 *B*2*,J* + … + *A1,N*  *BN,J*, *I* = 1, …, *M*, *J* = 1, …, *P*.

Квадратная матрица *A* порядка *M* называется *верхнетреугольной*, если все ее элементы, лежащие ниже главной диагонали, равны нулю:

*AI,J* = 0, *I* > *J*.

Квадратная матрица *A* порядка *M* называется *нижнетреугольной*, если все ее элементы, лежащие выше главной диагонали, равны нулю:

*AI,J* = 0, *I* < *J*.

Квадратная матрица *A* порядка *M* называется *трехдиагональной*, если равны нулю все ее элементы, не лежащие на главной диагонали и на двух диагоналях, примыкающих к главной:

*AI,J* = 0, | *I* – *J* | > 1.

1. Даны два целых числа *I*, *J* и файл вещественных чисел, содержащий элементы квадратной матрицы (по строкам). Вывести элемент матрицы, расположенный в *I*-й строке и *J*-м столбце (строки и столбцы нумеруются от 1). Если требуемый элемент отсутствует, то вывести 0.

| PascalABC.NET | |
| --- | --- |
| **Var**  F: **file of** Real;  Cell: Real;  I, J, i2, len: integer;  Name: String;  **begin**  Write('File: ');  Readln(Name);  Write('I: ');  Readln(I);  Write('J: ');  Readln(J);  Assign(F, Name);  Reset(F);  len := 0;  **while** (**not** eof(F)) **do**  **begin**  Read(F, Cell);  inc(len);  **end**;  len := trunc(sqrt(len));  Close(F);  Reset(F);  **for** i2 := 1 **to** len \* (i - 1) + j **do**  Read(F, Cell);  Writeln(Cell);  Close(F);  **end**. | |
| Python | C++ |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  i=int(input("i->"))  j=int(input("j->"))  n=0  for e in s:  n+=1  if n==i:  break  print(e.split(" ")[j-1]) | #include <iostream>  #include <vector>  #include <string>  #include <sstream>  #include <fstream>  using namespace std;  vector<string> split(string str,  char d) {  vector<string> v;  stringstream ss(str);  string tok;  while (getline(ss, tok, d)) {  v.push\_back(tok);  }  return v;  }  int main() {  ifstream ifs("D:/input.txt");  string S;  int i, j, k=0;  cin >> i >> j;  while (ifs) {  getline(ifs, S);  if (k == i-1)  break;  k++;  }  vector<string> sep=split(S, ' ');  cout<<sep[j-1]<<" "<<endl;  system("pause");  return 0;  } |
| Go | |
| package main  import (  "bufio"  "fmt"  "os"  "strings"  )  func main() {  f, \_ := os.Open("input.txt")  defer f.Close()  var i, j int  var n int=0  var s string  fmt.Scan(&i,&j)  scanner := bufio.NewScanner(f)  for scanner.Scan() {  line := scanner.Text()  n++  if n>=i{  s=line  break  }  }  fmt.Println(strings.Split(s, " ")[j-1])  } | |

# **Текстовые файлы: группа Text**

Условие «дан текстовый файл» означает, что в наборе исходных данных указано *имя* данного файла (текстовая строка). Все исходные файлы в заданиях данной группы считаются существующими.

Если в задании требуется создать новый файл, то имя создаваемого файла также входит в набор исходных данных (и, как правило, является последним элементом этого набора).

Максимальный размер исходных файлов не устанавливается, поэтому при решении заданий не следует использовать вспомогательные массивы, содержащие все элементы исходных файлов, однако допускается использование *вспомогательных файлов*.

Используемые в заданиях двоичные файлы удовлетворяют условиям, которые перечислены в начале раздела «Двоичные (типизированные) файлы».

## **Основные операции с текстовыми файлами**

1. Дано имя файла и целые положительные числа *N* и *K*. Создать текстовый файл с указанным именем и записать в него *N* строк, каждая из которых состоит из *K* символов «\*» (звездочка).

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  F: Text;  i, j, N, K: integer;  Name: String;  **begin**  Write('File name: ');  Readln(Name);  Write('N,K: ');  Readln(N, K);  Assign(F, Name);  ReWrite(F);  **for** i := 1 **to** N **do**  **begin**  **for** j := 1 **to** K **do**  F.Write('\*');  **if** i <> N **then**  F.Writeln;  **end**;  F.Close;  **end**. | | #include <iostream>  #include <string>  #include <fstream>  using namespace std;  int main() {  ofstream ofs  ("D:/output.txt");  string S;  int n, k, i;  cin >> n >> k;  S = S.append(k, '\*');  for (i = 0; i < n; i++) {  ofs << S << "\n";  }  system("pause");  return 0;  } | |
| Python | C# | | |
| f =open("D:/out.txt","w")  n=int(input("n->"))  k=int(input("k->"))  for i in range(0,n):  S=""  for j in range(0, k):  S=S+"\*"  S=S+"\n"  f.write(S)  f.close() | static void Main(string[] args) {  string fileName = @"D:/out.txt";  StreamWriter sw = new StreamWriter(fileName);  int n, k, i;  n=Convert.ToInt32(Console.ReadLine());  k = Convert.ToInt32(Console.ReadLine());  string S=new string('\*',k);  for (i=0; i<n; i++)  sw.WriteLine(S);  sw.Close();  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) throws IOException{  FileWriter writer = new FileWriter("D:\\output.txt");  Scanner sc=new Scanner(System.in);  int n, k, i;  n=sc.nextInt();  k=sc.nextInt();  String S="";  for (i=0; i<k; i++)  S=S+"\*";  for (i=0; i<n; i++)  writer.write(S+"\n");  writer.close();  } | | | package main  import (  "bufio"  "os"  "fmt"  )  func main() {  var k, n int  fmt.Scan(&k)  fmt.Scan(&n)  var s string=""  for i:=1; i<=k; i++{  s=s+"\* "  }  s=s+string("\n")  f, \_ :=  os.Create("file.txt")  w := bufio.NewWriter(f)  for i:=1; i<=n; i++{  w.WriteString(s)  }  w.Flush()  } |
| Kotlin | | | |
| **import** java.util.Scanner **import** java.io.File **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = reader.next()  **val** myfile = File(fileName)  **var** n = reader.nextInt()  **var** k = reader.nextInt()  **var** S = **""  for** (i **in** (1..k))  S = S + **"\*"**;  myfile.*printWriter*().*use* **{** out **->  for** (i **in** (1..n))  out.println(S)  **}** } | | | |

1. Дана строка *S* и текстовый файл. Добавить строку *S* в конец файла.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  F: Text;  Name, S: String;  **begin**  Write('File name: ');  Readln(Name);  Write('S: ');  Readln(S);  Assign(F, Name);  Append(F);  F.Writeln(S);  F.Close;  **end**. | | #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main()  {  string S;  getline(cin, S);  ofstream ofs("D:\input.txt", ios::app);  ofs << S;  return 0;  } | |
| Python | C# | | |
| S=input()  with open("D:\hello.txt", "a") as file:  file.write(S) | static void Main(string[] args)  {  string fileName = @"D:/out.txt";  StreamWriter sw = new StreamWriter  (fileName,true);  String S=Console.ReadLine();  sw.WriteLine(S);  sw.Close();  Console.ReadKey();  } | | |
| Java | | | Go |
| public static void main(String[] args) **throws IOException**{  FileWriter writer = new FileWriter("D:\\output.txt", true);  Scanner sc=new Scanner(System.in);  String S=sc.nextLine();  writer.write(S);  writer.close();  } | | | package main  import (  "fmt"  "os"  )  func main() {  f, err := os.OpenFile("file.txt", os.O\_APPEND|os.O\_WRONLY, os.ModeAppend)  if err != nil {  fmt.Println(err)  }  defer f.Close()  var s string  fmt.Scan(&s)  \_, err = f.WriteString(s)  if err != nil {  fmt.Println(err)  }  } |
| Kotlin | | | |
| **import** java.util.Scanner **import** java.io.File **import** java.io.FileWriter **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = reader.nextLine()  **val** myfile = File(fileName)  **var** S=reader.nextLine()  **val** fw = FileWriter(myfile, **true**)  fw.write(S)  fw.close() } | | | |
| Считать файл по строкам | | | |
| **1)**  **import** java.util.Scanner **import** java.io.File **import** java.io.FileWriter **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = reader.nextLine()  File(fileName).*readLines*().*forEach* **{** *println*(**it**)  **}** }  2)  **import** java.util.Scanner **import** java.io.File **import** java.io.FileWriter **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = reader.nextLine()  **val** file = File(fileName)  **val** bufferedReader = file.*bufferedReader*()  **val** text:List<String> = bufferedReader.*readLines*()  **for**(line **in** text){  *println*(line)  } }  3)  **import** java.util.Scanner **import** java.io.File **import** java.io.FileWriter **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **var** fileName = reader.nextLine()  **val** file = File(fileName)  **var** content:String = file.*readText*()  *println*(content) } | | | |

1. Дан текстовый файл. Продублировать в нем все пустые строки.

| PascalABC.NET | | C++ |
| --- | --- | --- |
| **Var**  F\_in, F\_out: Text;  Name, S: String;  **begin**  Write('File name: ');  Readln(Name);  Assign(F\_in, Name);  Assign(F\_out, 'D:\input1.txt');  Reset(F\_in);  Rewrite(F\_out);  **while not** F\_in.Eof **do**  **begin**  S := F\_in.ReadString;  **if** S = '' **then**  F\_out.Writeln;  F\_out.Writeln(S);  **end**;  F\_in.Close;  F\_out.Close;  Erase(F\_in);  F\_out.Rename(Name);  **end**. | | #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main()  {  string file1 = "D:\input.txt";  string file2 = "D:\input1.txt";  string S;  ifstream ifs(file1);  ofstream ofs(file2);  while (!ifs.eof()) {  getline(ifs, S);  if (S.length()== 0)  ofs << S << endl;  ofs << S << endl;  }  ifs.close();  ofs.close();  remove(file1.c\_str());  rename(file2.c\_str(),  file1.c\_str());  system("pause");  return 0;  } |
| Python | C# | |
| f=open("D:/input.txt","r")  s=f.readlines()  f.close()  f=open("D:/input.txt","w")  for e in s:  if e==" ":  f.write(e+"\n")  f.write(e+"\n")  f.close() | static void Main(string[] args)  {  string fileName = @"D:/out.txt";  StreamReader sr = new StreamReader  (fileName);  string line;  List<string> list = new List  <string>();  while((line=sr.ReadLine()) !=  null) {  if (line.Length==0)  list.Add("\n\n");  else  list.Add(line);  }  sr.Close();  StreamWriter sw = new StreamWriter  (fileName,false);  foreach (string x in list){  sw.WriteLine(x);  Console.WriteLine(x);  }  sw.Close();  Console.ReadKey();  } | |
| Java | | |
| public static void main(String[] args) **throws IOException**{  StringBuilder contentBuilder = new StringBuilder();  BufferedReader br = new BufferedReader(new FileReader  ("D:\\output.txt"));  String sCurrentLine;  while ((sCurrentLine = br.readLine()) != null){  if (sCurrentLine.isEmpty())  contentBuilder.append("\n\n");  else  contentBuilder.append(sCurrentLine).append("\n");  }  br.close();  FileWriter writer = new FileWriter("D:\\output.txt", false);  writer.write(contentBuilder.toString());  writer.close();  } | | |
| Go | | |
| package main  import (  "bufio"  "fmt"  "os"  )  func main() {  file, err := os.Create("file.txt")  if err != nil{  fmt.Println(err)  os.Exit(1)  }  defer file.Close()  f, \_ := os.Open("input.txt")  defer f.Close()  scanner := bufio.NewScanner(f)  for scanner.Scan() {  line := scanner.Text()  fmt.Println(line)  if len(line)==0{  fmt.Fprintf( file,"%s \n",line)  }  fmt.Fprintf( file,"%s \n",line)  }  os.Rename("file.txt","input.dat")  os.Remove("file.txt")  } | | |

1. Даны два текстовых файла. Добавить в конец каждой строки первого файла соответствующую строку второго файла. Если второй файл короче первого, то оставшиеся строки первого файла не изменять.

| PascalABC.NET |
| --- |
| **Var**  F1\_in, F2\_in, F\_out: Text;  Name1, Name2, line: string;  **begin**  Write('File1 name: ');  Readln(Name1);  Write('File2 name: ');  Readln(Name2);  Assign(F1\_in, Name1);  Assign(F2\_in, Name2);  Assign(F\_out, 'D:\temp.txt');  Reset(F1\_in);  Reset(F2\_in);  Rewrite(F\_out);  **while not** F1\_in.eof **do**  **begin**  line := F1\_in.ReadString;  F\_out.Write(line);  **if not** F2\_in.eof **then**  line := F2\_in.ReadString  **else**  line := '';  F\_out.Writeln(line);  **end**;  F1\_in.Close;  F2\_in.Close;  F\_out.Close;  F1\_in.Erase;  F\_out.Rename(Name1);  **end**. |
| Python |
| f1 = open("D:/input.txt","r")  s1 = f1.readlines()  f2 = open("D:/input1.txt","r")  s2 = f2.readlines()  f1.close()  f2.close()  f = open("D:/input2.txt","w")  for i in range(0, len(s1)):  if i<len(s2):  f.write(s1[i]+s2[i])  else:  f.write(s1[i])  f.close() |
| Go |
| package main  import (  "bufio"  "fmt"  "os"  )  func main() {  f1, \_ := os.Open("input.txt")  f2, \_ := os.Open("input1.txt")  f, \_ := os.Create("file.txt")  defer f1.Close()  defer f2.Close()  var s,s1,s2 string  scanner1 := bufio.NewScanner(f1)  scanner2 := bufio.NewScanner(f2)  for scanner1.Scan() {  s1 = scanner1.Text()  if scanner2.Scan() {  s2 = scanner2.Text()  }else{  s2=""  }  s=s1+" "+s2  fmt.Fprintf( f,"%s \n",s)  }  os.Remove("input.txt")  os.Rename("file.txt","input.txt")  os.Remove("file.txt")  } |

## **Анализ и форматирование текста**

1. Дан текстовый файл. Найти количество абзацев в тексте, если абзацы отделяются друг от друга одной или несколькими пустыми строками.

| PascalABC.NET | C++ |
| --- | --- |
| **var**  F: Text;  S: string;  num: integer;  **begin**  Write('File name: ');  Readln(S);  Assign(F, S);  Reset(F);  num := 0;  S := '';  **while** (**not** F.eof) **and** (S='') **do**  S := F.ReadlnString;  **while not** F.eof **do**  **begin**  S := F.ReadlnString;  **if** S = '' **then**  **begin**  inc(num);  **while** (**not** F.eof) **and** (S='') **do**  S := F.ReadlnString;  **end**;  **end**;  **if** S <> '' **then** inc(num);  F.Close;  Writeln(num);  **end**. | #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main()  {  string file1 = "D:\input.txt";  string S;  ifstream ifs(file1);  int k = 0;  while (!ifs.eof()) {  getline(ifs, S);  if (S.length()!= 0)  k++;  }  cout << k << endl;  system("pause");  return 0;  } |
| Python | |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  k=0  for e in s:  if e!="\n":  k+=1  print(k) | |
| Go | |
| package main  import (  "bufio"  "fmt"  "os"  )  func main() {  f, \_ := os.Open("input.txt")  defer f.Close()  var num int=0  scanner := bufio.NewScanner(f)  for scanner.Scan() {  s:= scanner.Text()  if (s!=""){  num++  }  }  fmt.Println(num)  } | |

1. Дан текстовый файл. Найти количество абзацев в тексте, если первая строка каждого абзаца начинается с 5 пробелов («*красная строка*»). Пустые строки между абзацами не учитывать.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  F: Text;  Name, S: string;  num, K: integer;  **begin**  Write('File name: ');  Readln(Name);  Assign(F, Name);  Reset(F);  num := 0;  **while not** F.eof **do**  **begin**  S := F.ReadlnString;  **if** pos(' ', S) = 1 **then**  inc(num);  **end**;  F.Close;  Writeln(num);  **end**. | #include <iostream>  #include <fstream>  #include <string>  using namespace std;  int main()  {  string file1 = "D:\input.txt";  string S;  ifstream ifs(file1);  int k = 0;  while (!ifs.eof()) {  getline(ifs, S);  if ((S.length()!= 0) &&  (S.find(" ") == 0))  k++;  }  cout << k << endl;  system("pause");  return 0;  } |
| Python | |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  k=0  for e in s:  if e.find(" ")==0:  k+=1  print(k) | |
| Go | |
| package main  import (  "bufio"  "fmt"  "os"  "strings"  )  func main() {  f, \_ := os.Open("input.txt")  defer f.Close()  var num int=0  scanner := bufio.NewScanner(f)  for scanner.Scan() {  s:= scanner.Text()  if strings.Index(s, " ")==0{  num++  }  }  fmt.Println(num)  } | |

1. Дан текстовый файл. Вывести первое слово текста наибольшей длины. *Словом* считать набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки.

| PascalABC.NET |
| --- |
| **Var**  F: Text;  Name, S, Word: string;  **begin**  Write('File name: ');  Readln(Name);  Assign(F, Name);  Reset(F);  Word := '';  **while not** F.eof **do**  **begin**  S := F.ReadlnString;  **while** (pos(' ', S) <> 0) **do**  **begin**  **if** (pos(' ', S) - 1) > Word.Length **then**  Word := Copy(S, 1, pos(' ', S) - 1)  **Else**  Delete(S, 1, pos(' ', S));  **end**;  **if** length(S) > Word.Length **then**  Word := S;  **end**;  F.Close;  Writeln(Word);  **end**. |
| Python |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  mx=0  word=""  for e in s:  a=e.split(" ")[0]  if len(a)>mx:  word=a  mx=len(a)  print(word) |
| Go |
| package main  import (  "bufio"  "fmt"  "os"  "strings"  )  func main() {  f, \_ := os.Open("input.txt")  defer f.Close()  var word string=" "  scanner := bufio.NewScanner(f)  for scanner.Scan() {  s:= scanner.Text()  t:=strings.Split(s, " ")[0]  if len(t)>len(word){  word=t  }  }  fmt.Println(word)  } |

1. Дан текстовый файл, содержащий текст, выровненный по левому краю. Выровнять текст по правому краю, добавив в начало каждой непустой строки нужное количество пробелов (ширину текста считать равной 50).

| PascalABC.NET |
| --- |
| **Var**  F\_in, F\_out: Text;  Name, S: string;  **begin**  Write('File name in: ');  Readln(Name);  Assign(F\_in, Name);  Assign(F\_out, 'D:\temp.txt');  Reset(F\_in);  Rewrite(F\_out);  **while not** F\_in.Eof **do**  **begin**  S := F\_in.ReadlnString;  **while** (S.Length <= 50) **do**  S := ' ' + S;  F\_out.Writeln(F\_out);  **end**;  F\_in.Close;  F\_out.Close;  F\_in.Erase;  F\_out.Rename(Name);  **end**. |
| Python |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  f = open("D:/input.txt","w")  for e in s:  if len(e)<=50:  while len(e)<50:  e=" "+e  f.write(e)  f.close() |
| Go |
| package main  import (  "bufio"  "fmt"  "os"  )  func main() {  f, \_ := os.Open("input.txt")  fo, \_ := os.Create("file.txt")  defer f.Close()  scanner := bufio.NewScanner(f)  for scanner.Scan() {  s:= scanner.Text()  if len(s)<=50{  for len(s)<52{  s=" "+s  }  }  fmt.Fprintf(fo,"%s \n",s)  }  } |

## **Текстовые файлы с числовой информацией**

В заданиях на обработку текстовых файлов с числовой информацией предполагается, что изображения вещественных чисел, содержащиеся в текстовых файлах, удовлетворяют стандартным правилам используемого языка программирования (в частности, дробная часть отделяется от целой части десятичной *точкой*).

1. Даны два файла целых чисел одинакового размера. Создать текстовый файл, содержащий эти числа, расположенные в два столбца шириной по 30 символов (в первом столбце содержатся числа из первого исходного файла, во втором — из второго файла). В начало и конец каждой строки текстового файла добавить разделитель «|» (код 124). Числа выравниваются по правому краю столбца.

| PascalABC.NET |
| --- |
| **Var**  F1\_in, F2\_in, F\_out: Text;  line, S, Name: string;  i: integer;  **begin**  Write('File 1 name in: ');  Readln(Name);  Assign(F1\_in, Name);  Write('File 2 name in: ');  Readln(Name);  Assign(F2\_in, Name);  Write('File name out: ');  Readln(Name);  Assign(F\_out, Name);  Reset(F1\_in);  Reset(F2\_in);  Rewrite(F\_out);  **while** (**not** eof(F1\_in)) **or** (**not** eof(F2\_in)) **do**  **begin**  line := '|';  **if** (**not** eof(F1\_in)) **then**  **begin**  read(F1\_in, i);  Str(i, S);  **end**  **else** S := '';  **for** i := 1 **to** 30 - Length(S) **do**  line := line + ' ';  line := line + S;  **if** (**not** eof(F2\_in)) **then**  **begin**  read(F2\_in, i);  Str(i, S);  **end**  **else** S := '';  **for** i := 1 **to** 30 - Length(S) **do**  line := line + ' ';  line := line + S;  line := line + '|';  Writeln(F\_out, line);  **end**;  Close(F1\_in);  Close(F2\_in);  Close(F\_out);  **end**. |
| Python |
| f = open("D:/input.txt","r")  s = f.readlines()  f.close()  x=[]  for e in s:  a=e.split(" ")  for i in range(0, len(a)-1):  x.append(a[i])  print(len(x))  f = open("D:/input1.txt","r")  s = f.readlines()  f.close()  y=[]  for e in s:  a=e.split(" ")  for i in range(0, len(a)-1):  y.append(a[i])  print(len(y))  f = open("D:/output.txt","w")  s=""  while len(s)<26:  s=s+" "  for i in range (0, len(x)-1):  s0="|"+x[i]+s+"|"+y[i]+s+"|\n";  f.write(s0)  f.close() |
| Go |
| package main  import (  "bufio"  "fmt"  "os"  "strings"  )  func main() {  f1, \_ := os.Open("input.txt")  f2, \_ := os.Open("input1.txt")  fo, \_ := os.Create("file.txt")  defer f1.Close()  defer f2.Close()  scanner1 := bufio.NewScanner(f1)  scanner2 := bufio.NewScanner(f2)  for scanner1.Scan() && scanner2.Scan() {  s1:=strings.Split(scanner1.Text()," ")  s2:=strings.Split(scanner2.Text()," ")  for i:=0; i<len(s2); i++{  s:=" "  for len(s)<30{  s=s+" "  }  so:="|"+s1[i]+s+"|"+s2[i]+s+"|";  fmt.Fprintf(fo,"%s \n", so)  }  }  } |

1. Даны вещественные числа *A*, *B* и целое число *N*. Создать текстовый файл, содержащий таблицу значений функции на промежутке [*A*, *B*]   
   с шагом . Таблица состоит из двух столбцов: с аргументами *x*   
   (10 позиций, из них 4 под дробную часть) и со значениями (15 позиций, из них 8 под дробную часть). Столбцы выравниваются по правому краю.

| PascalABC.NET | C++ |
| --- | --- |
| **Var**  F\_out: Text;  line, S, Name: string;  A, B, N, i: integer;  y, x: Real;  **begin**  Write('A: ');  Readln(A);  Write('B: ');  Readln(B);  Write('N: ');  Readln(N);  Write('File name out: ');  Readln(Name);  Assign(F\_out, Name);  Rewrite(F\_out);  y := A;  x := 0;  **while** x <= B **do**  **begin**  Write(F\_out, '|');  str(trunc(x), S);  **for** i:=1 **to** 5 - S.Length **do**  F\_out.Write(' ');  Write(F\_out, x:5:4);  y := sqrt(x);  str(trunc(y), S);  **for** i:= 1 **to** 6 - S.Length **do**  F\_out.Write(' ');  Write(F\_out, y:6:8);  F\_out.Writeln('|');  x := x + (B - A) / N;  **end**;  F\_out.Close;  **end**. | #include <iostream>  #include <fstream>  #include <string>  #include <cmath>  using namespace std;  int main()  {  string file1 = "D:\input.txt";  ofstream ofs(file1);  float x, y, a, b, n, h;  string S;  cin >> a >> b >> n;  h = abs(b - a) / n;  x = a;  S.append(6, ' ');  while (x < b) {  y = sqrt(x);  ofs << "|"<<S<<x<<"|"<<S+'  '<<y<<"|"<<endl;  x+=h;  }  ofs.close();  system("pause");  return 0;  } |
| Python | |
| import math  f = open("D:/output.txt","w")  s=""  while len(s)<6:  s=s+" "  a=float(input("a->"))  b=float(input("b->"))  n=int(input("n->"))  h=abs(b-a)/n  x=a  while x<b:  y=math.sqrt(x)  s0="|"+str(x)+s+"|"+str(y)+s+" |\n"  f.write(s0)  print(s0)  x+=h  f.close() | |
| Go | |
| package main  import (  "fmt"  "os"  "math"  )  func main() {  var x,y,h,a,b,n float64  fmt.Scan(&a, &b,&n)  h=math.Abs(b-a)/n  fo, \_ := os.Create("file.txt")  defer fo.Close()  for x=a; x<=b; x+=h{  y=math.Sqrt(x)  s:=" "  for len(s)<6{  s=s+" "  }  so:= fmt.Sprintf("%f", x)+s  s=" "  for len(s)<6{  s=s+" "  }  so= so+fmt.Sprintf("%f", y)+s  fmt.Fprintf(fo,"%s \n", so)  fmt.Println(so)  }  } | |

## **Дополнительные задания на обработку текстовых файлов**

1. Дан текстовый файл. Создать символьный файл, содержащий все знаки препинания, встретившиеся в текстовом файле (в том же порядке).

| PascalABC.NET |
| --- |
| **Var**  F\_in: Text;  F\_out: **file of** Char;  Name: string;  C: Char;  **begin**  Write('File In name: ');  Readln(Name);  Assign(F\_in, Name);  Reset(F\_in);  Write('File Out name: ');  Readln(Name);  Assign(F\_out, Name);  Rewrite(F\_out);  **while not** F\_in.eof **do**  **begin**  **while not** F\_in.eoln **do**  **begin**  C := F\_in.ReadChar;  **if** (char.IsPunctuation(C)) **then**  F\_out.Write(C);  **end**;  F\_in.Readln;  **end**;  F\_in.Close;  F\_out.Close;  **end**. |
| Python |
| f = open("D:/input.txt","r+")  s=f.readlines()  f.close()  punctuations = '''!()-[]{};:'"\,<>./?@#$%^&\*\_~'''  f = open("D:/output.txt","w")  for e in s:  for i in range(0, len(e)-1):  if e[i] in punctuations:  f.write(e[i]) |
| Go |
| package main  import (  "bufio"  "fmt"  "os"  "unicode"  )  func main() {  f, \_ := os.Open("input.txt")  fo, \_ := os.Create("file.txt")  defer f.Close()  defer fo.Close()  scanner := bufio.NewScanner(f)  for scanner.Scan() {  s := scanner.Text()  for i:=0; i<len(s); i++{  if unicode.IsPunct(rune(s[i])){  fmt.Fprintf(fo,"%c ",rune(s[i]))  }  }  }  } |

**Задачи для самостоятельной работы**

**Вариант №1**

1. Заполнить файл последовательного доступа *f* целыми числами, полученными с помощью генератора случайных чисел. Получить в файле *g* те компоненты файла *f*, которые являются четными. Багаж пассажира характеризуется количеством вещей и общим весом вещей.
2. Дан файл Bagazh, содержащий сведения о багаже нескольких пассажиров. Сведения о багаже каждого пассажира представляют собой запись с двумя полями: одно поле целого типа (количество вещей) и одно — действительное (вес в килограммах). Найти багаж, средний вес одной вещи в котором отличается не более чем на 0,3 кг от общего среднего веса одной вещи.
3. Дан файл, содержащий текст, записанный строчными русскими буквами. Получить в другом файле тот же текст, записанный заглавными буквами.

**Вариант №2**

1. Записать в файл последовательного доступа *N* действительных чисел. Вычислить произведение компонент файла и вывести на печать.
2. Багаж пассажира характеризуется количеством вещей и общим весом вещей. Дан файл Bagazh, содержащий сведения о багаже нескольких пассажиров. Сведения о багаже каждого пассажира представляют собой запись с двумя полями: одно поле целого типа (количество вещей) и одно — действительное (вес в килограммах). Найти число пассажиров, имеющих более двух вещей и число пассажиров, количество вещей которых превосходит среднее число вещей.
3. Дан файл, содержащий произвольный текст. Выяснить, чего в нем больше: русских букв или цифр.

**Вариант №3**

1. Заполнить файл последовательного доступа *f* целыми числами, полученными с помощью генератора случайных чисел. Получить в файле g все компоненты файла *f*, которые делятся на *m* и не делятся на *n*.
2. Багаж пассажира характеризуется количеством вещей и общим весом вещей. Дан файл **Bagazh**, содержащий сведения о багаже нескольких пассажиров, Сведения о багаже каждого пассажира представляют собой запись с двумя полями: одно поле целого типа (количество вещей) и одно - действительное (вес в килограммах). Выяснить, имеется ли пассажир, багаж которого состоит из одной вещи весом менее 30 кг.
3. Дан файл, содержащий текст на русском языке. Выяснить, входит ли данное слово в указанный текст, и если да, то сколько раз.

**Вариант №4**

1. Записать в файл последовательного доступа *N* целых чисел, полученных с помощью генератора случайных чисел. Подсчитать количество пар противоположных чисел среди компонент этого файла.
2. Дан файл **Bibl**, содержащий сведения о книгах. Сведения о каждой из книг — это фамилия автора, название и год издания. Найти название книг данного автора, изданных с I960 года.
3. Дан файл, содержащий текст на русском языке. В предложениях некоторые из слов записаны подряд несколько раз (предложение заканчивается точкой или знаком восклицания). Получить в новом файле отредактированный текст, в котором удалены повторные вхождения слов в предложение.

**Вариант №5**

1. Заполнить файл последовательного доступа *f* целыми числами, полученными с помощью генератора случайных чисел. Из файла *f* получить файл g, исключив повторные вхождения чисел. Вывести файл g на печать.
2. Дан файл **Bibl**, содержащий сведения о книгах. Сведения о каждой иp книг — это фамилия автора, название и год издания. Определить, имеется ли книга с названием "Информатика". Если да, то на печатать фамилию автора и год издания. Если таких книг несколько, то напечатать имеющиеся сведения обо всех этих книгах.
3. Дан файл, содержащий текст, набранный заглавными русскими буквами. В нем содержатся также знаки препинания. Провести частотный анализ текста, т.е. указать (в процентах) сколько раз встречается та или иная буква.

**Вариант №6**

1. Записать в файл последовательного доступа *N* произвольных натуральных чисел. Переписать в другой файл последовательного доступа те элементы, которые кратны *K*. Вывести полученный файл на печать.
2. Дан файл **Т**, который содержит номера телефонов сотрудников учреждения: указывается фамилия, инициалы и номер телефона. Найти номер телефона сотрудника по его фамилии и инициалам.
3. Дан файл, содержащий текст на русском языке. Определить, сколько раз встречается в нем самое длинное слово.

**Вариант №7**

1. Заполнить файл последовательного доступа *N* действительными числами, полученными с помощью датчика случайных чисел. Найти сумму минимального и максимального элементов этого файла.
2. Дан файл, содержащий различные даты. Каждая дата — это число месяц и год. Найти год с наименьшим номером.
3. Дан файл, содержащий произвольный текст. Проверить, правильно ли в нем расставлены круглые скобки (т.е. находится ли правее каждой открывающейся скобки закрывающаяся, и левее закрывающейся — открывающаяся).

**Вариант №8**

1. Записать в файл последовательного доступа *N* натуральных чисел: *a*1, *a*2, …, *an* (числа, получить с помощью датчика случайных чисел). Сформировать новый файл последовательного доступа, элементами которого являются числа: *a1, a1*×*a2*, *a1*×*a2*×*a3*, …, *a1*×*a2*×*a3*×…×*an*.
2. Дан файл, содержащий различные даты. Каждая дата — это число месяц и год. Найти все весенние даты.
3. Дан файл, содержащий текст на русском языке. Составить в алфавитном порядке список всех слов, встречающихся в этом тексте.

**Вариант №9**

1. Записать в файл *f* последовательного доступа *N* натуральных чисел. Получить в другом файле последовательного доступа все компоненты файла *f*, кроме тех, которые кратны *K*. Вывести полученный файл на печать.
2. Дан файл, содержащий различные даты. Каждая дата — это число месяц и год. Найти самую позднюю дату.
3. Дан файл, содержащий текст на русском языке. Определить, сколько раз встречается в нем самое короткое слово.

**Вариант №10**

1. Заполнить файл *f* целыми числами, полученными с помощью генератора случайных чисел. Найти количество удвоенных нечетных чисел среди компонент файла.
2. Дан файл **Tovar**, содержащий сведения об экспортируемых товарах: указывается наименование товара, страна, импортирующая товар, и объем поставляемой партии в штуках. Составить список стран, в которые экспортируется данный товар, и общий объем его экспорта.
3. Дан файл, содержащий текст на русском языке и некоторые два слова. Определить, сколько раз они входят в текст и сколько раз они входят непосредственно друг за другом.

**Вариант №11**

1. Заполнить файл *f* натуральными числами, полученными с помощью генератора случайных чисел. Найти количество квадратов нечетных чисел среди компонент.
2. Дан файл **Assort**, содержащий сведения об игрушках: указывается название игрушки, ее стоимость в рублях и возрастные границы (например, игрушка может предназначаться для детей от двух до пяти лет). Получить название игрушек, цена которых не превышает 14 тыс. р. и которые подходят детям 5 лет.
3. Дан файл, содержащий текст на русском языке. Выбрать из него только те символы, которые встречаются в нем только один раз, в том порядке, в котором они встречаются в тексте.

**Вариант №12**

1. Записать в файл прямого доступа *N* действительных чисел. Найти наибольшее из значений модулей компонент с нечетными номерами.
2. Дан файл **Assort**, содержащий сведения об игрушках: указывается название игрушки, ее стоимость в рублях и возрастные границы (например, игрушка может предназначаться для детей от двух до пяти лет). Определить стоимость самого дорогого конструктора.
3. Дан файл, содержащий текст и арифметические выражения вида *a∙ b* где *∙* — один из знаков +, -, \*, /. Выписать все арифметические выражения и вычислить их значения.

**Вариант №13**

1. Заполнить файл f целыми числами, полученными с помощью генератора случайных чисел. Из файла f получить файл g, исключив повторные вхождения чисел. Порядок следования чисел сохранить.
2. Дан файл **Assort**, содержащий сведения об игрушках: указывается название игрушки, ее стоимость в рублях и возрастные границы (например, игрушка может предназначаться для детей от двух до пяти лет). Напечатать название наиболее дорогих игрушек (цена которых отличается от цены самой дорогой игрушки не более, чем на 5 тыс. р.).
3. Дан файл, содержащий текст на русском языке и некоторая буква. Найти слово, содержащее наибольшее количество указанных букв.

**Вариант №14**

1. Записать в файл последовательного доступа *N* действительных чисел. Найти разность первой и последней компонент файла.
2. Дан файл **Assort**, содержащий сведения об игрушках: указывается название игрушки, ее стоимость в рублях и возрастные границы (например, игрушка может предназначаться для детей от двух до пяти лет). Получить названия игрушек, которые подходят детям как четырех лет, таи и десяти лет.
3. Дан файл, содержащий текст на русском языке и некоторая буква. Подсчитать, сколько слов начинается с указанной буквы.

**Вариант №15**

1. Записать в файл *f* *N* целых чисел, полученных с помощью генератора случайных чисел. Заполнить файл g числами, которые являются произведениями соседних компонент файла *f*.
2. Дан фалл **Assort**, содержащий сведения об игрушках: указывается название игрушки, ее стоимость в рублях и возрастные границы (например, игрушка может предназначаться для детей от двух до пяти лет). Получить сведения о том, можно ли подобрать игрушку; любую, кроме мяча, подходящую ребенку трех лет.
3. Дан файл, содержащий текст на русском языке. Найти слово, встречающееся в каждом предложении, или сообщить, что такого слова нет.

**Вариант №16**

1. Записать в файл последовательного доступа *n* элементов последовательности *bn* = 1-1/2!+1/3!-1/4!+…+.(-1)*n*-1×1/*n*!. Вывести на печать те компоненты файла, для которых выполняется |*bn*| > ε, где ε - заданное число.
2. Дан файл **Assort**, содержащий сведения об игрушках: указывается название игрушки, ее стоимость в рублях и возрастные границы (например, игрушка может предназначаться для детей от двух до пяти лет). Получить сведения о том, можно ли подобрать игрушки так, чтобы суммарная стоимость игрушек не превосходила 50 тыс. р.
3. Дан файл, содержащий текст, включающий русские и английские слова. Подсчитать, каких букв в тексте больше — русских или латинских.

**Вариант №17**

1. Записать в файл последовательного доступа *N* действительных чисел *a*1, *a*2, …, *an*. Организовать новый файл последовательного доступа, элементы которого вычисляются по формуле:  Вывести полученный файл на печать.
2. Дан файл Assort, содержащий сведения об игрушках: укалывается название игрушки, ее стоимость в рублях и возрастные границы (например, игрушка может предназначаться для детей от двух до пяти лет). Получить название самой дешевой игрушки.
3. Дан файл, содержащий текст. Сколько слов в тексте? Сколько цифр в тексте?

**Вариант 18**

1. Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные после первого символа пробела, включая и этот пробел.
2. Дан файл вещественных чисел. Заменить в нем все элементы на их квадраты.
3. Дан файл целых чисел, содержащий не менее четырех элементов. Вывести первый, второй, предпоследний и последний элементы данного файла.

**Вариант 19**

1. Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные после последнего символа пробела, включая и этот пробел.
2. Дан файл вещественных чисел. Поменять в нем местами минимальный и максимальный элементы.
3. Дан файл целых чисел. Создать новый файл, содержащий те же элементы, что и исходный файл, но в обратном порядке.

**Вариант** 20

1. Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные перед первым символом пробела, включая и этот пробел.
2. Дан файл целых чисел с элементами *A*1, *A*2, …, *AN* (*N* — количество элементов в файле). Заменить исходное расположение его элементов на следующее:

*A*1,    *AN*,    *A*2,    *AN*–1,    *A*3,    … .

1. Дан файл вещественных чисел. Создать два новых файла, первый из которых содержит элементы исходного файла с нечетными номерами (1, 3, …), а второй — с четными (2, 4, …).

**Вариант 21**

1. Дан символьный файл, содержащий по крайней мере один символ пробела. Удалить все его элементы, расположенные перед последним символом пробела, включая и этот пробел.
2. Дан файл вещественных чисел. Заменить в файле каждый элемент, кроме начального и конечного, на его среднее арифметическое с предыдущим и последующим элементом.
3. Дан файл целых чисел. Создать два новых файла, первый из которых содержит четные числа из исходного файла, а второй — нечетные (в том же порядке). Если четные или нечетные числа в исходном файле отсутствуют, то соответствующий результирующий файл оставить пустым.

**Вариант 22**

1. Дан символьный файл. Упорядочить его элементы по возрастанию их кодов.
2. Дан файл целых чисел, содержащий более 50 элементов. Уменьшить его размер до 50 элементов, удалив из файла необходимое количество конечных элементов.
3. Дан файл целых чисел. Создать два новых файла, первый из которых содержит положительные числа из исходного файла (в обратном порядке), а второй — отрицательные (также в обратном порядке). Если положительные или отрицательные числа в исходном файле отсутствуют, то соответствующий результирующий файл оставить пустым.

**Вариант 23**

1. Дано целое число *K* (> 0) и строковый файл. Создать два новых файла: строковый, содержащий первые *K* символов каждой строки исходного файла, и символьный, содержащий *K*-й символ каждой строки (если длина строки меньше *K*, то в строковый файл записывается вся строка, а в символьный файл записывается пробел).
2. Дан файл целых чисел, содержащий четное количество элементов. Удалить из данного файла вторую половину элементов.
3. Дан файл вещественных чисел. Найти среднее арифметическое его элементов.

**Вариант 24**

1. Дан строковый файл. Создать новый строковый файл, содержащий все строки исходного файла наименьшей длины (в том же порядке).
2. Дан файл целых чисел, содержащий более 50 элементов. Уменьшить его размер до 50 элементов, удалив из файла необходимое количество начальных элементов.
3. Дан файл вещественных чисел. Найти сумму его элементов с четными номерами.

**Вариант 25**

1. Дан строковый файл. Создать новый строковый файл, содержащий все строки исходного файла наибольшей длины (в обратном порядке).
2. Дан файл целых чисел, содержащий четное количество элементов. Удалить из данного файла первую половину элементов.
3. Дан файл целых чисел. Найти количество содержащихся в нем *серий* (то есть наборов последовательно расположенных одинаковых элементов). Например, для файла с элементами 1, 5, 5, 5, 4, 4, 5 результат равен 4.

# **Составные типы данных в процедурах и функциях: группа Param**

В каждом задании данного раздела требуется описать процедуру или функцию и затем использовать ее для обработки исходных данных.

Все параметры любой *функции* считаются входными. Для *процедур* всегда указывается, какие параметры являются выходными (или одновременно входными и выходными); если о виде параметра процедуры ничего не сказано, то он считается входным.

## **Одномерные и двумерные массивы**

При вводе исходного массива вначале следует ввести его размер (одно число для одномерных массивов, два числа — количество строк и столбцов — для двумерных массивов-матриц), а затем — все его элементы.

Если в задании явно не указывается размер одномерного массива, являющегося параметром процедуры или функции, то предполагается, что этот размер может меняться в пределах от 1 до 10. Для двумерных массивов-матриц также предполагается, что число их строк и столбцов может меняться от 1 до 10. Порядковые номера начальных элементов как одномерных, так и двумерных массивов всегда считаются равными 1.

При описании процедур, выполняющих преобразование массива, не следует использовать вспомогательный массив того же размера

1. Описать функцию MinInt(*A*, *N*) целого типа, находящую минимальный элемент целочисленного массива *A* размера *N*. С помощью этой функции найти минимальные элементы массивов *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  A,B,C: **array of** integer;  Na, Nb, Nc: integer;  **function** MinInt(a: **array of** integer; N: integer): Integer;  **var**  MinEl, i: integer;  **begin**  MinEl := 1;  **for** i := 1 **to** N **do**  **if** (A[i]<A[MinEl]) **then**  MinEl := i;  MinInt := MinEl;  **end**;  **procedure** EnterArray(a: **array of** integer; **var** N: integer);  **var**  i: integer;  **begin**  **for** i := 1 **to** N **do**  A[i] := random(50);  **end**;  **procedure** PrintArray(a: **array of** integer; **var** N: integer);  **var**  i: integer;  **begin**  **for** i := 1 **to** N **do**  print(a[i]);  writeln;  **end**;  **begin**  Writeln('Enter N:');  Na := ReadInteger;  a := **new** integer[Na + 1];  EnterArray(A, Na);  PrintArray(A, Na);  Writeln(MinInt(A, Na));  Writeln('Enter N:');  Nb := ReadInteger;  B := **new** integer[Nb + 1];  EnterArray(B, Nb);  PrintArray(B, Nb);  Writeln(MinInt(B, Nb));  Writeln('Enter N:');  Nc := ReadInteger;  C := **new** integer[Nc + 1];  EnterArray(C, Nc);  PrintArray(C, Nc);  Writeln(MinInt(C, Nc));  **end**. | | #include <iostream>  using namespace std;  int minInt(int x[], int n) {  int mn = x[0];  for (int i = 1; i < n; i++) {  if (x[i] < mn)  mn = x[i];  }  return mn;  }  int \*init (int n) {  int \*x = new int[n];  for (int i = 0; i < n; i++) {  x[i] = rand() % 50;  }  return x;  }  void printArray(int a[], int n) {  for (int i = 0; i < n; i++)  cout << a[i] << " ";  }  int main(){  int \*a,\*b,\*c,Na, Nb, Nc, i;  cin >> Na >> Nb >> Nc;  a = init(Na);  b = init(Nb);  c = init(Nc);  printArray(a, Na);  cout<<"\n"<<minInt(a,Na)<< endl;  printArray(b, Nb);  cout<<"\n"<<minInt(b,Nb)<<endl;  printArray(c, Nc);  cout<<"\n"<<minInt(c,Nc)<<endl;  system("pause");  return 0;  } | |
| Python | C# | | |
| def minInt(a,n):  a=[]  for i in range (0, n):  a.append(int(input()))  return min(a)  n=int(input("n->"))  a=[]  print (minInt(a,n)) | static int MinInt(int[] x){  int mn = x[0];  for (int i = 1; i < x.Length; i++)  if (mn > x[i])  mn = x[i];  return mn;  }  static int[] init(int n){  int[] x = new int[n];  Random rnd = new Random();  for (int i = 0; i < n; i++)  x[i] = rnd.Next(n, n\*10);  PrintArray(x);  return x;  }  static void PrintArray(int[]x){  for (int i = 0; i < x.Length; i++)  Console.Write(x[i] + " ");  Console.WriteLine();  }  static void Main(string[] args) {  int Na, Nb, Nc;  Na = Convert.ToInt32(Console.ReadLine());  Nb = Convert.ToInt32(Console.ReadLine());  Nc= Convert.ToInt32(Console.ReadLine());  int[] a = init(Na);  int[] b = init(Nb);  int[] c = init(Nc);  Console.WriteLine(MinInt(a) + " " + MinInt(b) + " " + MinInt(c));  Console.ReadKey();  } | | |
| Java | | | Go |
| static int MinInt(int[]x){  int mn=x[0];  for (int i=1; i<x.length; i++)  if (mn>x[i])  mn=x[i];  return mn;  }  static int []init(int n){  int []x=new int[n];  Random rnd=new Random();  for (int i=0; i<n; i++)  x[i]=rnd.nextInt(30);  PrintArray(x);  return x;  }  static void PrintArray(int []x){  for (int i=0; i<x.length; i++)  System.out.print(x[i]+" ");  System.out.println();  }  public static void main(String[] args){  int Na,Nb,Nc;  Scanner sc=new Scanner(System.in);  Na=sc.nextInt();  Nb=sc.nextInt();  Nc=sc.nextInt();  int []a=init(Na);  int []b=init(Nb);  int []c=init(Nc);  System.out.println(MinInt(a)+" "+MinInt(b)+" "+MinInt(c));  } | | | package main  import (  "fmt"  "math/rand"  )  func MinInt(a[] byte)byte{  var min=a[0]  for i:=1; i<len(a); i++{  if min>a[i]{  min=a[i]  }  }  return min  }  func main() {  var Na,Nb, Nc int  fmt.Println("Na, Nb, Nc->")  fmt.Scan(&Na, &Nb, &Nc)  a := make([]byte, Na)  rand.Read(a)  fmt.Println(a)  fmt.Println(MinInt(a))  b := make([]byte, Nb)  rand.Read(b)  fmt.Println(b)  fmt.Println(MinInt(b))  c := make([]byte, Nc)  rand.Read(c)  fmt.Println(c)  fmt.Println(MinInt(c))  } |
| Kotlin | | | |
| **import** java.util.Scanner **fun** IntMin(x: Array<Int>, n:Int):Int{  **var** mn=x[0];  **for** (i **in** (1..n-1)){  **if** (x[i]<mn)  mn=x[i]  }  **return** mn } **fun** main(args: Array<String>) {  **val** reader = Scanner(System.*`in`*)  **val** n = reader.nextInt();  **val** x = Array(n,**{**i**->**2\*i+1**}**)  **for**(i **in** 0..n-1) {  *println*(x[i])  }  *print*(*IntMin*(x,n)) } | | | |
| VB.NET | | | |
| Function minInt(x() As Integer, n As Integer) As Integer  Dim min As Integer = x(0)  For i As Integer = 1 To n - 1  If min > x(i) Then  min = x(i)  End If  Next  Return min  End Function  Sub Main()  Dim i, n, min As Integer  n = Convert.ToInt32(Console.ReadLine())  Dim a() As Integer = New Integer(n) {}  Dim rnd As Random = New Random()  For i = 0 To n - 1  a(i) = rnd.Next(50)  Console.Write("{0} ", a(i))  Next  Console.WriteLine()  min = minInt(a, n)  Console.WriteLine(min)  Console.ReadKey()  End Sub | | | |

1. Описать функцию Norm1(*A*, *M*, *N*) вещественного типа, вычисляющую *норму* вещественной матрицы *A* размера M×N:

Norm1(*A*, *M*, *N*) = max {|*A*1,*J*| + |*A*2,*J*| + … + |*AM*,*J*|},

где максимум берется по всем *J* от 1 до *N*. Для данной матрицы *A* размера M×N найти Norm1(*A*, *K*, *N*), *K* = 1, …, *M*.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Var**  M, N, K: integer;  A: **array**[,] **of** integer;  **function** Norm1(**var** A: **array**[,] **of** integer; M: integer; N: integer): Real;  **begin**  **var** NormMax := 0;  **for var** i := 1 **to** M **do**  **begin**  **var** Norm := 0;  **for var** j := 1 **to** N **do**  Norm := Norm + a[i, j];  **if** (Norm>NormMax) **or** (i=1) **then**  NormMax := Norm;  **end**;  Norm1 := NormMax;  **end**;  **procedure** InitMatrix(**var** B: **array**[,] **of** integer; **var** M, N: integer);  **begin**  **for var** i := 1 **to** M **do**  **begin**  **for var** j := 1 **to** N **do**  B[i, j] := random(50);  **end**;  Writeln;  **end**;  **procedure** PrintMatrix(**var** B: **array**[,] **of** integer; **var** M, N: integer);  **begin**  **for var** i := 1 **to** M **do**  **begin**  **for var** j := 1 **to** N **do**  print(B[i, j]);  Writeln;  **end**;  Writeln;  **end**;  **begin**  Write('M: ');  Readln(M);  Write('N: ');  Readln(N);  A:=**new** integer[M + 1, N + 1];  InitMatrix(A, M, N);  PrintMatrix(A, M, N);  **for** K := 1 **to** M **do**  Writeln(Norm1(A, K, N));  **end**. | | #include <iostream>  using namespace std;  void fill\_array(int \*\*arr, int m, int n) {  int count = 0;  for (int i = 0; i<m; i++) {  for (int j = 0; j<n; j++) {  arr[i][j] = rand()%30;  }  }  }  void print\_array(int\*\* arr, int m,  int n) {  for (int i = 0; i < m; i++) {  for (int j = 0; j < n; j++)  cout << arr[i][j] << '\t';  cout << '\n';  }  cout << '\n';  }  int Norm1(int \*\*x, int m, int n) {  int NormMax = 0;  for (int i = 0; i < m; i++) {  int Norm = 0;  for (int j = 0; j < n; j++) {  Norm = Norm + x[i][j];  if ((Norm>NormMax) || (i == 0))  NormMax = Norm;  }  }  return NormMax;  }  int main() {  int m, n;  cout << "m, n: ";  cin >> m >> n;  int \*\*x = new int\*[m];  for (int i = 0; i<m; i++) {  x[i] = new int[n];  }  fill\_array(x, m, n);  print\_array(x, m, n);  cout << Norm1(x, m, n) << endl;  for (int i = 0; i<m; i++) {  delete[]x[i];  }  delete[]x;  system("pause");  return 0;  } | |
| Python | C# | | |
| import random  def init(c, m, n):  for i in range(m):  for j in range(n):  c[i][j] = random.randrange(0,101, 5)  return c  def Norm1(c, m, n):  NormMax=0  for i in range(m):  Norm=0  for j in range(m):  Norm=Norm+c[i][j];  if Norm>NormMax or i==0:  NormMax = Norm  return NormMax  def show(a, m, n):  init(a, 5, 5)  for i in range(m):  for j in range(m):  print(a[i][j], end=' ')  print()  a = [[2]\*5 for \_ in range(5)]  show(a, 5,5)  print(Norm1(a,5,5)) | static int Norm1(int[,]x, int m, int n){  int NormMax=0;  for (int i = 0; i < m; i++) {  int Norm=0;  for (int j = 0; j < n; j++) {  Norm = Norm + x[i,j];  if ((Norm>NormMax) || (i==0))  NormMax = Norm;  }  }  return NormMax;  }  static int [,]init(int m, int n){  int [,]x=new int[m,n];  Random rnd=new Random();  for (int i=0; i<m; i++)  for (int j=0; j<n; j++)  x[i,j]=rnd.Next(50);  PrintArray(x, m, n);  return x;  }  static void PrintArray(int [,]x, int m, int n){  for (int i=0; i<m; i++){  for (int j=0; j<n; j++)  Console.Write(x[i,j]+" ");  Console.WriteLine();  }  Console.WriteLine();  }  static void Main(string[] args) {  int m, n;  m=Convert.ToInt32(Console.ReadLine());  n=Convert.ToInt32(Console.ReadLine());  int[,]a=init(m,n);  Console.WriteLine(Norm1(a, m, n));  Console.ReadKey();  } | | |
| Java | | | Go |
| static int Norm1(int[][]x,int m,int n){  int NormMax=0;  for (int i = 0; i < m; i++) {  int Norm=0;  for (int j = 0; j < n; j++) {  Norm = Norm + x[i][j];  if ((Norm>NormMax) || (i==0))  NormMax = Norm;  }  }  return NormMax;  }  static int [][]init(int m, int n){  int [][]x=new int[m][n];  Random rnd=new Random();  for (int i=0; i<m; i++)  for (int j=0; j<n; j++)  x[i][j]=rnd.nextInt(30);  PrintArray(x, m, n);  return x;  }  static void PrintArray(int [][]x, int m, int n){  for (int i=0; i<m; i++){  for (int j=0; j<n; j++)  System.out.print(x[i][j]+" ");  System.out.println();  }  System.out.println();  }  public static void main(String[] args){  int m,n;  Scanner sc=new Scanner(System.in);  m=sc.nextInt();  n=sc.nextInt();  int[][]a=init(m,n);  System.out.println(Norm1(a, m, n));  } | | | package main  import (  "fmt"  "math/rand"  )  func print2D(arr [][]int) {  for i:=0;i<len(arr); i++ {  forj:=0;j<len(arr[0]);j++ { fmt.Printf("%d ",arr[i][j])  }  fmt.Println()  }  }  func Norm1(arr [][]int) int{  NormMax:=0  for i:=0; i<len(arr);i++{  Norm:=0  for j:=0;j<len(arr[0]);j++ {  Norm = Norm + arr[i][j];  if Norm>NormMax || i==0{  NormMax = Norm  }  }  }  return NormMax  }  func main() {  var arr [][]int  var m int  fmt.Println("m->")  fmt.Scan(&m)  for i := 0; i < m; i++ {  tmp := make([]int, m)  for j := 0; j < m; j++ {  tmp[j]=rand.Intn(50)  }  arr = append(arr, tmp)  }  print2D(arr)  fmt.Println(Norm1(arr))  } |

## **Строки**

1. Описать функцию IsIdent(*S*) целого типа, проверяющую, является ли строка *S* допустимым *идентификатором*, то есть непустой строкой, которая содержит только латинские буквы, цифры и символ подчеркивания «\_» и не начинается с цифры. Если *S* является допустимым идентификатором, то функция возвращает 0. Если *S* является пустой строкой, то возвращается –1, если *S* начинается с цифры, то возвращается –2. Если *S* содержит недопустимые символы, то возвращается номер первого недопустимого символа. Проверить с помощью функции IsIdent пять данных строк.

| PascalABC.NET |
| --- |
| **Var**  S: String;  i: integer;  **function** IsIdent(S: String): integer;  **var**  i, temp: integer;  **begin**  temp := 0;  **if** length(s) > 0 **then**  **if** (s[1] **in** ['a'..'z', '\_']) **then**  **begin**  **for** i := length(S) **downto** 2 **do**  **if not** (s[i] **in** ['0'..'9', 'a'..'z', '\_']) **then**  temp := i;  **end**  **else**  temp := -2  **else**  temp := -1;  IsIdent := temp;  **end**;  **begin**  **for** i := 1 **to** 5 **do**  **begin**  Write('S:');  Readln(S);  Writeln(IsIdent(S));  **end**;  **end**. |
| Python |
| def IsIndent(S):  temp=0  if len(S)>0:  if S[0] in [range(ord('a'), ord('z')), '\_']:  for i in range (len(S), 1, -1):  if S[i] not in [range(ord('a'), ord('z')),  range(0,9), '\_']:  temp=i  else:  temp=-2  else:  temp=-1  return temp  for i in range (0, 5):  S=input("S->")  print(IsIndent(S)) |

1. Описать функцию CompressStr(*S*) строкового типа, выполняющую *сжатие* строки *S* по следующему правилу: каждая подстрока строки *S*, состоящая из более чем четырех одинаковых символов *C*, заменяется текстом вида «*С*{*K*}», где *K* — количество символов *C* (предполагается, что строка *S* не содержит фигурных скобок «{» и «}»). Например, для строки *S* = «bbbccccce» функция вернет строку «bbbc{5}e». С помощью функции CompressStr сжать пять данных строк.

| PascalABC.NET |
| --- |
| **Var**  S: String;  i: integer;  **function** CompressStr(S: String): String;  **var**  i, i2, Num: integer;  Stemp: String;  C: Char;  **begin**  C := S[1];  Num := 1;  Result := '';  **for** i := 2 **to** length(S) **do**  **begin**  **if** S[i] = C **then** inc(num);  **if** (S[i] <> C) **or** (i = length(S)) **then**  **begin**  **if** num > 3 **then**  **begin**  Str(num, Stemp);  Result := Result + C + '{' + Stemp + '}';  **End**  **else**  **for** i2 := 1 **to** num **do**  Result := Result + C;  C := S[i];  Num := 1;  **end**;  **end**;  **end**;  **begin**  **for** i := 1 **to** 5 **do**  **begin**  Write('S:');  Readln(S);  Writeln(CompressStr(S));  **end**;  **end**. |
| Python |
| def CompressStr(S):  C=S[0]  Num=1  Rezult=""  for i in range (1, len(S)):  if S[i]==C:  Num+=1  if S[i]!=C or i==len(S)-1:  if Num>3:  Rezult=Rezult+C+"{"+str(Num)+"}"  else:  for j in range (0, Num):  Rezult = Rezult + C;  C=S[i]  Num=1  return Rezult  for i in range (0, 5):  S=input("S->")  print(CompressStr(S)) |
| Go |
| package main  import (  "fmt"  "strconv"  )  func CompressStr(S string) string{  C:=S[0]  Num:=1  Rezult:=""  for i:=1; i<len(S); i++{  if S[i]==C{  Num+=1  }  if S[i]!=C || i==len(S)-1{  if Num>3{  Rezult=Rezult+string(C)+"{"+strconv.Itoa(Num)+"}"  }else{  for j:=0; j<Num; j++{  Rezult = Rezult + string(C)  }  }  C=S[i]  Num=1  }  }  return Rezult  }  func main() {  var S string  fmt.Scan(&S)  fmt.Println(CompressStr(S))  } |

## **Файлы**

1. Описать процедуру AddLineNumbers(*S*, *N*, *K*, *L*), добавляющую в начало каждой строки существующего текстового файла с именем *S* ее порядковый номер: первая строка получает номер *N*, вторая *N* + 1 и т. д. Номер отображается в *K* позициях, выравнивается по правому краю и отделяется от последующего текста *L* пробелами (*K* > 0, *L* > 0). Если строка файла является пустой, то она также нумеруется, но пробелы после номера не добавляются. Применить эту процедуру к данному файлу, используя указанные значения *N*, *K* и *L*.

| PascalABC.NET |
| --- |
| **Type**  FileName = String;  **var**  N, K, L: integer;  S: FileName;  **procedure** AddLineNumbers(S: FileName; N, K, L: integer);  **var**  F, F\_temp: Text;  i: integer;  TempS, Line: String;  **begin**  Assign(F, S);  Reset(F);  Assign(F\_temp, 'temp.txt');  ReWrite(F\_temp);  **while** (**not** F.eof) **do**  **begin**  TempS := '';  str(N, TempS);  **for** i := TempS.Length **to** K - 1 **do**  TempS := ' ' + TempS;  Line := F.ReadlnString;  **if** Line.Length > 0 **then**  **for** i := 1 **to** L **do**  TempS := TempS + ' ';  TempS := TempS + Line;  F\_temp.Writeln(TempS);  inc(N);  **end**;  F.Close;  F\_temp.Close;  F.Erase;  F\_temp.Rename(S);  **end**;  **begin**  Write('S: ');  Readln(S);  Write('N: ');  Readln(N);  Write('K: ');  Readln(K);  Write('L: ');  Readln(L);  AddLineNumbers(S, N, K, L);  **end**. |
| Python |
| def FileRead(S,n,k,l):  f=open(S,"r")  s=f.readlines()  f.close()  S0=""  S1=""  for i in range(0, k):  S0=S0+" "  for i in range(0, l):  S1=S1+" "  num=n  f=open(S,"w")  for e in s:  if len(e)<=1:  SS=S0+str(num)  else:  SS=S0+str(num)+S1+e  num+=1  f.write(SS)  n=int(input("n->"))  k=int(input("k->"))  l=int(input("l->"))  FileRead("D:/input.txt",n,k,l); |
| Go |
| package main  import (  "bufio"  "fmt"  "log"  "os"  )  func FileRead(S string,n,k,l int){  file, err := os.Open(S)  if err != nil {  log.Fatal(err)  }  S0:=""  S1:=""  for i:=0;i<k; i++{  S0=S0+" "  }  for i:=0; i<l; i++{  S1=S1+" "  }  num:=n  fileo, err1:= os.Create("D:/temp.txt")  if err1 != nil{  fmt.Println(err1)  os.Exit(1)  }  scanner := bufio.NewScanner(file)  for scanner.Scan() {  var SS string=""  e:=scanner.Text()  if len(e)<=1{  SS=S0+string(num)  }else{  SS=S0+string(num)+S1+e  }  num+=1  fmt.Fprintf( fileo,"%s \n", SS)  }  file.Close()  fileo.Close()  os.Rename("D:/temp.txt",S)  }  func main() {  var n,k,l int  fmt.Scan(&n,&k,&l)  FileRead("D:/input.txt",n,k,l);  } |

## **Записи**

При вводе и выводе каждой даты вначале указывается день, затем номер месяца, затем год. При вводе каждой точки вначале указывается ее абсцисса (*x*-координата), затем ее ордината (*y*-координата).

1. Описать тип TDate — запись с полями целого типа Day (день), Month (месяц) и Year (год) — и функцию LeapYear(*D*) логического типа с параметром типа TDate, которая возвращает True, если год в дате *D* является високосным, и False в противном случае. Вывести значение функции LeapYear для пяти данных дат (предполагается, что все даты являются правильными). *Високосным* считается год, делящийся на 4, за исключением тех годов, которые делятся на 100 и не делятся на 400.

| PascalABC.NET | | C++ | |
| --- | --- | --- | --- |
| **Type**  TDate = **record**  Day: integer;  Month: integer;  Year: integer;  **end**;  **var**  D: TDate;  **function** LeaP\_Year(D: TDate): boolean;  **begin**  **if** (D.Year **mod** 100) = 0 **then**  **if** (D.Year **mod** 400) = 0 **then**  LeaP\_Year := True  **else** LeaP\_Year := False  **else**  **if** (D.Year **mod** 4) = 0 **then**  LeaP\_Year := True  **else**  LeaP\_Year := False;  **end**;  **begin**  Write('Date: ');  Readln(D.Day, D.Month, D.Year);  Writeln(LeaP\_Year(D));  **end**. | | struct TDate {  public: int Day, Month, Year;  };  bool LeaP\_Year(TDate D) {  if (D.Year % 100 == 0)  return D.Year % 400 == 0;  else return D.Year % 4 == 0;  }  int main(){  TDate d;  cin>>d.Day>>d.Month>>d.Year;  cout << LeaP\_Year(d);  system("pause");  return 0;  } | |
| Python | C# | | |
| class TDate(object):  def \_\_init\_\_(self, Day, Month, Year):  self.Day = Day  self.Month = Month  self.Year = Year  def LeaP\_Year(D: TDate):  if D.Year % 100 == 0:  if D.Year % 400 ==0:  return True  else:  return False  else:  if D.Year % 4 == 0:  return True  else:  return False  x=TDate(int(input()), int(input()), int(input()))  print(LeaP\_Year(x)) | struct TDate{  public int Day, Month, Year;  };  static bool LeaP\_Year(TDate D){  if (D.Year % 100 == 0)  return D.Year % 400 == 0;  else  return D.Year % 4 == 0;  }  static void Main(string[] args) {  TDate d;  d.Day=Convert.ToInt32(Console.ReadLine()); d.Month=Convert.ToInt32(Console.ReadLine());  d.Year=Convert.ToInt32(Console.ReadLine());  Console.WriteLine(LeaP\_Year(d));  Console.ReadKey();  } | | |
| Java | | | Go |
| static boolean LeaP\_Year(TDate D){  if (D.Year % 100 == 0){  return D.Year % 400 ==0;  }  else  return D.Year % 4 == 0;  }  public static void main(String[] args) {  TDate d=new TDate();  Scanner sc=new Scanner(System.in);  d.Day=sc.nextInt();  d.Month=sc.nextInt();  d.Year=sc.nextInt();  System.out.println(LeaP\_Year(d));  } | | | package main  import "fmt"  type TData struct {      Day int      Month int      Year int  }  func LeaP\_Year(D TData) bool{  if D.Year % 100 == 0{  if D.Year % 400 ==0{  return true  }else{  return false  }  }else{  if D.Year % 4 == 0{  return true  } else{  return false  }  }  }  func main() {  var d = TData{25,4, 2000}  fmt.Println(LeaP\_Year(d))  } |

# **Задачи для самостоятельной работы**

В каждом задании данного раздела требуется описать процедуру или функцию и затем использовать ее для обработки исходных данных.

Все параметры любой *функции* считаются входными. Для *процедур* всегда указывается, какие параметры являются выходными (или одновременно входными и выходными); если о виде параметра процедуры ничего не сказано, то он считается входным.

**Одномерные и двумерные массивы**

1. Описать функцию NMax(*A*, *N*) целого типа, находящую номер максимального элемента вещественного массива *A* размера *N*. С помощью этой функции найти номера максимальных элементов массивов *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно.
2. Описать процедуру NMinmax(*A*, *N*, *NMin*, *NMax*), находящую номера минимального и максимального элемента вещественного массива *A* размера *N*. Выходные параметры целого типа: *NMin* (номер минимального элемента) и *NMax* (номер максимального элемента). С помощью этой процедуры найти номера минимальных и максимальных элементов массивов *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно.
3. Описать процедуру Invert(*A*, *N*), меняющую порядок следования элементов вещественного массива *A* размера *N* на противоположный (*инвертирование* массива). Массив *A* является входным и выходным параметром. С помощью этой процедуры инвертировать массивы *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно.
4. Описать процедуру Smooth1(*A*, *N*), выполняющую *сглаживание* вещественного массива *A* размера *N* следующим образом: элемент *AK* заменяется на среднее арифметическое первых *K* исходных элементов массива *A*. Массив *A* является входным и выходным параметром. С помощью этой процедуры выполнить пятикратное сглаживание данного массива *A* размера *N*, выводя результаты каждого сглаживания.
5. Описать процедуру Smooth2(*A*, *N*), выполняющую *сглаживание* вещественного массива *A* размера *N* следующим образом: элемент *A*1 не изменяется, элемент *AK* (*K* = 2, …, *N*) заменяется на полусумму исходных элементов *AK*–1 и *AK*. Массив *A* является входным и выходным параметром. С помощью этой процедуры выполнить пятикратное сглаживание данного массива *A* размера *N*, выводя результаты каждого сглаживания.
6. Описать процедуру Smooth3(*A*, *N*), выполняющую *сглаживание* вещественного массива *A* размера *N* следующим образом: каждый элемент массива заменяется на его среднее арифметическое с соседними элементами (при вычислении среднего арифметического используются *исходные* значения соседних элементов). Массив *A* является входным и выходным параметром. С помощью этой процедуры выполнить пятикратное сглаживание данного массива *A* размера *N*, выводя результаты каждого сглаживания.
7. Описать процедуру RemoveX(*A*, *N*, *X*), удаляющую из целочисленного массива *A* размера *N* элементы, равные целому числу *X*. Массив *A* и число *N* являются входными и выходными параметрами. С помощью этой процедуры удалить числа *XA*, *XB*, *XC* из массивов *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно и вывести размер и содержимое полученных массивов.
8. Описать процедуру RemoveForInc(*A*, *N*), удаляющую из вещественного массива *A* размера *N* «лишние» элементы так, чтобы оставшиеся элементы оказались упорядоченными по возрастанию: первый элемент не удаляется, второй элемент удаляется, если он меньше первого, третий — если он меньше предыдущего элемента, оставленного в массиве, и т. д. Например, массив 5.5, 2.5, 4.6, 7.2, 5.8, 9.4 должен быть преобразован к виду 5.5, 7.2, 9.4. Массив *A* и число *N* являются входными и выходными параметрами. С помощью этой процедуры преобразовать массивы *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно и вывести размер и содержимое полученных массивов.
9. Описать процедуру DoubleX(*A*, *N*, *X*), дублирующую в целочисленном массиве *A* размера *N* элементы, равные целому числу *X*. Массив *A* и число *N* являются входными и выходными параметрами. С помощью этой процедуры продублировать числа *XA*, *XB*, *XC* в массивах *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно и вывести размер и содержимое полученных массивов.
10. Описать процедуру Sort(*A*, *N*), выполняющую сортировку по возрастанию вещественного массива *A* размера *N*. Массив *A* является входным и выходным параметром. С помощью этой процедуры отсортировать массивы *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно.
11. Описать процедуру SortIndex(*A*, *N*, *I*), формирующую для вещественного массива *A* размера *N* *индексный массив* *I* — массив целых чисел того же размера, содержащий номера элементов массива *A* в том порядке, который соответствует возрастанию элементов массива *A* (сам массив *A* при этом не изменяется). Индексный массив *I* является выходным параметром. С помощью этой процедуры создать индексные массивы для массивов *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно.
12. Описать процедуру Bell(*A*, *N*), меняющую порядок элементов вещественного массива *A* размера *N* на следующий: наименьший элемент массива располагается на первом месте, наименьший из оставшихся элементов — на последнем, следующий по величине располагается на втором месте, следующий — на предпоследнем и т. д. (в результате график значений элементов будет напоминать *колокол*). Массив *A* является входным и выходным параметром. С помощью этой процедуры преобразовать массивы *A*, *B*, *C* размера *NA*, *NB*, *NC* соответственно.
13. Описать процедуру Split2(*A*, *NA*, *B*, *NB*, *C*, *NC*), формирующую по целочисленному массиву *A* размера *NA* два целочисленных массива *B* и *C* размера *NB* и *NC* соответственно; при этом массив *B* содержит все четные числа из массива *A*, а массив *C* — все нечетные числа (в том же порядке). Массивы *B* и *C* и числа *NB* и *NC* являются выходными параметрами. Применить эту процедуру к данному массиву *A* размера *NA* и вывести размер и содержимое полученных массивов *B* и *C*.
14. Описать процедуру ArrayToMatrix1(*A*, *K*, *M*, *N*, *B*), формирующую по вещественному массиву *A* размера *K* матрицу *B* размера M×N (матрица заполняется элементами массива *A* по строкам). «Лишние» элементы массива игнорируются; если элементов массива недостаточно, то оставшиеся элементы матрицы полагаются равными 0. Двумерный массив *B* является выходным параметром. С помощью этой процедуры на основе данного массива *A* размера *K* и целых чисел *M* и *N* сформировать матрицу *B* размера M×N.
15. Описать процедуру ArrayToMatrix2(*A*, *K*, *M*, *N*, *B*), формирующую по вещественному массиву *A* размера *K* матрицу *B* размера M×N (матрица заполняется элементами массива *A* по столбцам). «Лишние» элементы массива игнорируются; если элементов массива недостаточно, то оставшиеся элементы матрицы полагаются равными 0. Двумерный массив *B* является выходным параметром. С помощью этой процедуры на основе данного массива *A* размера *K* и целых чисел *M* и *N* сформировать матрицу *B* размера M×N.
16. Описать процедуру Chessboard(*M*, *N*, *A*), формирующую по целым положительным числам *M* и *N* матрицу *A* размера M×N, которая содержит числа 0 и 1, расположенные в «шахматном» порядке, причем *A*1,1 = 0. Двумерный целочисленный массив *A* является выходным параметром. С помощью этой процедуры по данным целым числам *M* и *N* сформировать матрицу *A* размера M×N.
17. Описать функцию Norm2(*A*, *M*, *N*) вещественного типа, вычисляющую *норму* вещественной матрицы *A* размера M×N:

Norm2(*A*, *M*, *N*) = max {|*AI*,1| + |*AI*,2| + … + |*AI*,*N*|},

где максимум берется по всем *I* от 1 до *M*. Для данной матрицы *A* размера M×N найти Norm2(*A*, *K*, *N*), *K* = 1, …, *M*.

1. Описать функцию SumRow(*A*, *M*, *N*, *K*) вещественного типа, вычисляющую сумму элементов вещественной матрицы *A* размера M×N, расположенных в *K*-й строке (если *K* > *M*, то функция возвращает 0). Для данной матрицы *A* размера M×N и трех данных *K* найти SumRow(*A*, *M*, *N*, *K*).
2. Описать функцию SumCol(*A*, *M*, *N*, *K*) вещественного типа, вычисляющую сумму элементов вещественной матрицы *A* размера M×N, расположенных в *K*-м столбце (если *K* > *N*, то функция возвращает 0). Для данной матрицы *A* размера M×N и трех данных *K* найти SumCol(*A*, *M*, *N*, *K*).
3. Описать процедуру SwapRow(*A*, *M*, *N*, *K*1, *K*2), осуществляющую перемену местами строк вещественной матрицы *A* размера M×N с номерами *K*1 и *K*2. Матрица *A* является входным и выходным параметром; если *K*1 или *K*2 больше *M*, то матрица не изменяется. Используя эту процедуру, поменять для данной матрицы *A* размера M×N строки с данными номерами *K*1 и *K*2.
4. Описать процедуру SwapCol(*A*, *M*, *N*, *K*1, *K*2), осуществляющую перемену местами столбцов вещественной матрицы *A* размера M×N с номерами *K*1 и *K*2. Матрица *A* является входным и выходным параметром; если *K*1 или *K*2 больше *N*, то матрица не изменяется. Используя эту процедуру, поменять для данной матрицы *A* размера M×N столбцы с данными номерами *K*1 и *K*2.
5. Описать процедуру Transp(*A*, *M*), выполняющую *транспонирование* (то есть зеркальное отражение относительно главной диагонали) квадратной вещественной матрицы *A* порядка *M*. Матрица *A* является входным и выходным параметром. Используя эту процедуру, транспонировать данную матрицу *A* порядка *M*.
6. Описать процедуру RemoveRows(*A*, *M*, *N*, *K*1, *K*2), удаляющую из вещественной матрицы *A* размера M×N строки с номерами от *K*1 до *K*2 включительно (предполагается, что 1 < *K*1 ≤ *K*2). Если *K*1 > *M*, то матрица не изменяется; если *K*2 > *M*, то удаляются строки матрицы с номерами от *K*1 до *M*. Двумерный массив *A* и числа *M*, *N* являются входными и выходными параметрами. Используя процедуру RemoveRows, удалить из данной матрицы *A* размера M×N строки с номерами от *K*1 до *K*2 и вывести размер полученной матрицы и ее элементы.
7. Описать процедуру RemoveCols(*A*, *M*, *N*, *K*1, *K*2), удаляющую из вещественной матрицы *A* размера M×N столбцы с номерами от *K*1 до *K*2 включительно (предполагается, что 1 < *K*1 ≤ *K*2). Если *K*1 > *N*, то матрица не изменяется; если *K*2 > *N*, то удаляются столбцы матрицы с номерами от *K*1 до *N*. Двумерный массив *A* и числа *M*, *N* являются входными и выходными параметрами. Используя процедуру RemoveCols, удалить из данной матрицы *A* размера M×N столбцы с номерами от *K*1 до *K*2 и вывести размер полученной матрицы и ее элементы.
8. Описать процедуру RemoveRowCol(*A*, *M*, *N*, *K*, *L*), удаляющую из вещественной матрицы *A* размера M×N строку и столбец, которые содержат элемент *AK*,*L* (предполагается, что *M* > 1 и *N* > 1; если *K* > *M* или *L* > *N*, то матрица не изменяется). Двумерный массив *A* и числа *M*, *N* являются входными и выходными параметрами. Дана матрица *A* размера M×N и числа *K*, *L*. Применить к матрице *A* процедуру RemoveRowCol и вывести размер полученной матрицы и ее элементы.
9. Описать процедуру SortCols(*A*, *M*, *N*), выполняющую сортировку по возрастанию столбцов целочисленной матрицы *A* размера M×N (столбцы сравниваются *лексикографически*: если первые элементы столбцов различны, то меньшим считается столбец, содержащий меньший первый элемент; если первые элементы столбцов равны, то анализируются их вторые элементы и т. д.). Двумерный массив *A* является входным и выходным параметром. Используя процедуру SortCols, отсортировать столбцы данной матрицы *A* размера M×N.

**Строки**

1. Описать функцию FillStr(*S*, *N*) строкового типа, возвращающую строку длины *N*, заполненную повторяющимися копиями строки-*шаблона* *S* (последняя копия строки-шаблона может входить в результирующую строку частично). Используя эту функцию, сформировать по данному числу *N* и пяти данным строкам-шаблонам пять результирующих строк длины *N*.
2. Описать процедуру UpCaseRus(*S*), преобразующую все строчные русские буквы строки *S* в прописные (остальные символы строки *S* не изменяются). Строка *S* является входным и выходным параметром. Используя процедуру UpCaseRus, преобразовать пять данных строк.
3. Описать процедуру LowCaseRus(*S*), преобразующую все прописные русские буквы строки *S* в строчные (остальные символы строки *S* не изменяются). Строка *S* является входным и выходным параметром. Используя процедуру LowCaseRus, преобразовать пять данных строк.
4. Описать процедуру TrimLeftC(*S*, *C*), удаляющую в строке *S* начальные символы, совпадающие с символом *C*. Строка *S* является входным и выходным параметром. Дан символ *C* и пять строк. Используя процедуру TrimLeftC, преобразовать данные строки.
5. Описать процедуру TrimRightC(*S*, *C*), удаляющую в строке *S* конечные символы, совпадающие с символом *C*. Строка *S* является входным и выходным параметром. Дан символ *C* и пять строк. Используя процедуру TrimRightC, преобразовать данные строки.
6. Описать функцию InvertStr(*S*, *K*, *N*) строкового типа, возвращающую *инвертированную подстроку* строки *S*, содержащую в обратном порядке *N* символов строки *S*, начиная с ее *K*-го символа. Если *K* превосходит длину строки *S*, то возвращается пустая строка; если длина строки меньше *K* + *N*, то инвертируются все символы строки, начиная с ее *K*-го символа. Вывести значения функции InvertStr для данной строки *S* и каждой из трех пар положительных целых чисел: (*K*1, *N*1), (*K*2, *N*2), (*K*3, *N*3).
7. Описать функцию PosSub(*S*0, *S*, *K*, *N*) целого типа, возвращающую номер позиции, начиная с которой в строке *S* содержится первое вхождение строки *S*0, причем анализируются только *N* символов строки *S*, начиная с ее *K*-го символа (таким образом, PosSub обеспечивает *поиск в подстроке*). Если *K* превосходит длину строки *S*, то возвращается 0, если длина строки меньше *K* + *N*, то анализируются все символы строки, начиная с ее *K*-го символа. Если в требуемой подстроке строки *S* вхождения *S*0 отсутствуют, то функция возвращает 0. Вывести значения функции PosSub для данных строк *S*0, *S* и каждой из трех пар положительных целых чисел: (*K*1, *N*1), (*K*2, *N*2), (*K*3, *N*3).
8. Описать функцию PosLast(*S*0, *S*) целого типа, возвращающую номер позиции, начиная с которой в строке *S* содержится последнее вхождение подстроки *S*0. Считать, что перекрывающихся вхождений подстрок *S*0 строка *S* не содержит. Если в строке *S* отсутствуют подстроки *S*0, то функция возвращает 0. Вывести значения этой функции для пяти данных пар строк *S*0 и *S*.
9. Описать функцию PosK(*S*0, *S*, *K*) целого типа, возвращающую номер позиции, начиная с которой в строке *S* содержится *K*-е вхождение подстроки *S*0 (*K* > 0). Если количество вхождений *S*0 в строке *S* меньше *K*, то функция возвращает 0. Считать, что перекрывающихся вхождений подстрок *S*0 строка *S* не содержит. Вывести значения этой функции для пяти данных троек: *S*0, *S* и *K*.
10. Описать функцию WordK(*S*, *K*) строкового типа, возвращающую *K*-е слово строки *S* (*словом* считается набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки). Если количество слов в строке меньше *K*, то функция возвращает пустую строку. Используя эту функцию, выделить из данной строки *S* слова с данными номерами *K*1, *K*2, *K*3.
11. Описать процедуру SplitStr(*S*, *W*, *N*), которая формирует по данной строке *S* массив *W* слов, входящих в *S* (массив *W* и его размер *N* являются выходными параметрами). *Словом* считается набор символов, не содержащий пробелов и ограниченный пробелами или началом/концом строки; предполагается, что строка *S* содержит не более 10 слов. Используя функцию SplitStr, найти количество слов *N*, содержащихся в данной строке *S*, и сами эти слова.
12. Описать функцию DecompressStr(*S*) строкового типа, восстанавливающую строку, сжатую процедурой CompressStr. Параметр *S* содержит сжатую строку; восстановленная строка является возвращаемым значением функции. С помощью функции DecompressStr восстановить пять данных сжатых строк.
13. Описать функцию DecToBin(*N*) строкового типа, возвращающую строковое представление целого неотрицательного числа *N* в двоичной системе счисления. Результирующая строка состоит из символов «0»–«1» и не содержит ведущих нулей (за исключением представления числа 0). Используя эту функцию, получить двоичные представления пяти данных чисел.
14. Описать функцию DecToHex(*N*) строкового типа, возвращающую строковое представление целого неотрицательного числа *N* в 16-ричной системе счисления. Результирующая строка состоит из символов «0»–«9», «A»–«F» и не содержит ведущих нулей (за исключением представления числа 0). Используя эту функцию, получить 16-ричные представления пяти данных чисел.
15. Описать функцию BinToDec(*S*) целого типа, определяющую целое неотрицательное число по его строковому представлению *S* в двоичной системе счисления. Параметр *S* имеет строковый тип, состоит из символов «0»–«1» и не содержит ведущих нулей (за исключением значения «0»). Используя эту функцию, вывести пять чисел, для которых даны их двоичные представления.
16. Описать функцию HexToDec(*S*) целого типа, определяющую целое неотрицательное число по его строковому представлению *S* в 16-ричной системе счисления. Параметр *S* имеет строковый тип, состоит из символов «0»–«9», «A»–«F» и не содержит ведущих нулей (за исключением значения «0»). Используя эту функцию, вывести пять чисел, для которых даны их 16-ричные представления.
17. Описать функцию IntFileSize(*S*) целого типа, возвращающую количество элементов в файле целых чисел с именем *S*. Если файл не существует, то функция возвращает –1. С помощью этой функции найти количество элементов в трех файлах с данными именами.

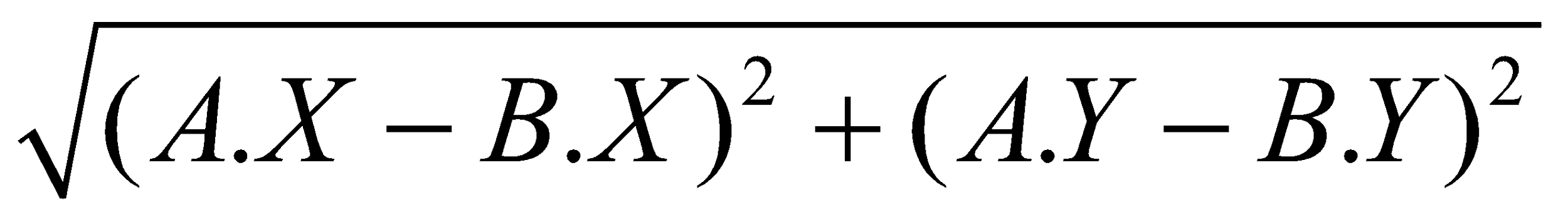
**Файлы**

1. Описать функцию LineCount(*S*) целого типа, возвращающую количество строк в текстовом файле с именем *S*. Если файл не существует, то функция возвращает –1. С помощью этой функции найти количество строк в трех файлах с данными именами.
2. Описать процедуру InvertIntFile(*S*), меняющую порядок следования элементов файла целого типа с именем *S* на противоположный. Если файл не существует или содержит менее двух элементов, то процедура не выполняет никаких действий. Обработать с помощью этой процедуры три файла с данными именами.
3. Описать процедуру RemoveLineNumbers(*S*), удаляющую из начала каждой строки существующего текстового файла с именем *S* ее порядковый номер, добавленный процедурой AddLineNumbers, а также пробелы, отделяющие номер от последующего текста. Если строки не содержат номеров, то процедура не выполняет никаких действий. Применить эту процедуру к файлу с данным именем.
4. Описать процедуру SplitIntFile(*S*0, *K*, *S*1, *S*2), копирующую первые *K* (> 0) элементов существующего файла целых чисел с именем *S*0 в новый файл целых чисел с именем *S*1, а остальные элементы — в новый файл целых чисел с именем *S*2. Один из созданных файлов может остаться пустым. Применить эту процедуру к файлу с данным именем *S*0, используя указанные значения *K*, *S*1 и *S*2.
5. Описать процедуру SplitText(*S*0, *K*, *S*1, *S*2), копирующую первые *K* () строк существующего текстового файла с именем *S*0 в новый текстовый файл с именем *S*1, a остальные строки — в новый текстовый файл с именем *S*2. Один из созданных файлов может остаться пустым. Применить эту процедуру к файлу с данным именем *S*0, используя указанные значения *K*, *S*1 и *S*2.
6. Описать процедуру StringFileToText(*S*), преобразующую двоичный строковый файл с именем *S* в текстовый файл с тем же именем. Используя эту процедуру, преобразовать два данных строковых файла с именами *S*1 и *S*2 в текстовые.
7. Описать процедуру TextToStringFile(*S*), преобразующую текстовый файл с именем *S* в двоичный строковый файл с тем же именем. Используя эту процедуру, преобразовать два данных текстовых файла с именами *S*1 и *S*2 в строковые.
8. Описать процедуру EncodeText(*S*, *K*), которая шифрует текстовый файл с именем *S*, выполняя циклическую замену каждой русской буквы на букву того же регистра, расположенную в алфавите на *K*-й позиции после шифруемой буквы (0 < *K* < 10). Например, при *K* = 3 «А» перейдет в «Г», «я» — в «в». Букву «ё» в алфавите не учитывать, считая, что за буквой «е» сразу идет «ж». Символы, не являющиеся русскими буквами, при шифровании не изменять. Используя эту процедуру и зная кодовое смещение *K*, зашифровать файл с указанным именем.
9. Описать процедуру DecodeText(*S*, *K*), которая дешифрует текстовый файл с именем *S*, зашифрованный с использованием кодового смещения *K*. Используя эту процедуру и зная кодовое смещение *K*, расшифровать файл с указанным именем.

**Записи**

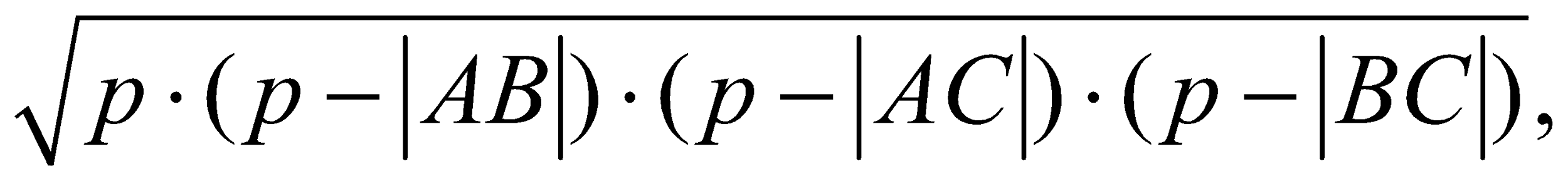
При вводе и выводе каждой даты вначале указывается день, затем номер месяца, затем год. При вводе каждой точки в заданиях Param64–Param70 вначале указывается ее абсцисса (*x*-координата), затем ее ордината (*y*-координата).

1. Используя тип TDate и функцию LeapYear, описать функцию DaysInMonth(*D*) целого типа с параметром типа TDate, которая возвращает количество дней для месяца, указанного в дате *D*. Вывести значение функции DaysInMonth для пяти данных дат (предполагается, что все даты являются правильными).
2. Используя тип TDate и функцию DaysInMonth, описать функцию CheckDate(*D*) целого типа с параметром типа TDate, которая проверяет правильность даты, указанной в параметре *D*. Если дата *D* является правильной, то функция возвращает 0; если в дате указан неверный номер месяца, то функция возвращает 1; если в дате указан неверный день для данного месяца, то возвращается 2. Вывести значение функции CheckDate для пяти данных дат.
3. Используя тип TDate и функции DaysInMonth и CheckDate, описать процедуру PrevDate(*D*) с параметром типа TDate, которая преобразует дату *D* к предыдущей дате (если дата *D* является неправильной, то она не изменяется). Запись *D* является входным и выходным параметром. Применить процедуру PrevDate к пяти данным датам.
4. Используя тип TDate и функции DaysInMonth и CheckDate, описать процедуру NextDate(*D*) с параметром типа TDate, которая преобразует дату *D* к следующей дате (если дата *D* является неправильной, то она не изменяется). Запись *D* является входным и выходным параметром. Применить процедуру NextDate к пяти данным датам.
5. Описать тип TPoint — запись с полями вещественного типа *X* и *Y* (координаты точки на плоскости) — и функцию Leng(*A*, *B*) вещественного типа, находящую длину отрезка *AB* на плоскости по координатам его концов:

|*AB*| = 

(*A* и *B* — параметры типа TPoint). С помощью этой функции найти длины отрезков *AB*, *AC*, *AD*, если даны координаты точек *A*, *B*, *C*, *D*.

1. Используя тип TPoint и функцию Leng, описать тип TTriangle — запись с полями *A*, *B*, *C* типа TPoint (вершины треугольника) — и функцию Perim(*T*) вещественного типа, находящую периметр треугольника *T* (*T* — параметр типа TTriangle). С помощью этой функции найти периметры треугольников *ABC*, *ABD*, *ACD*, если даны координаты точек *A*, *B*, *C*, *D*.
2. Используя типы TPoint, TTriangle и функции Leng и Perim, описать функцию Area(*T*) вещественного типа, находящую площадь треугольника *T* (*T* — параметр типа TTriangle) по *формуле Герона*:

*SABC* = 

где *p* — *полупериметр*. С помощью этой функции найти площади треугольников *ABC*, *ABD*, *ACD*, если даны координаты точек *A*, *B*, *C*, *D*.

1. Используя типы TPoint, TTriangle и функции Leng и Area, описать функцию Dist(*P*, *A*, *B*) вещественного типа (*P*, *A*, *B* — параметры типа TPoint), находящую расстояние *D*(*P*, *AB*) от точки *P* до прямой *AB* по формуле

*D*(*P*, *AB*) = 2·*SPAB*/|*AB*|,

где *SPAB* — площадь треугольника *PAB*. С помощью этой функции найти расстояния от точки *P* до прямых *AB*, *AC*, *BC*, если даны координаты точек *P*, *A*, *B*, *C*.

1. Используя типы TPoint, TTriangle и функцию Dist, описать процедуру Heights(*T*, *h*1, *h*2, *h*3), находящую высоты *h*1, *h*2, *h*3 треугольника *T* (*T* — входной параметр типа TTriangle, *h*1, *h*2, *h*3 — выходные вещественные параметры), проведенные соответственно из вершин *T*.*A*, *T*.*B*, *T*.*C*. С помощью этой процедуры найти высоты треугольников *ABC*, *ABD*, *ACD*, если даны координаты точек *A*, *B*, *C*, *D*.
2. Используя тип TPoint и функцию Leng, описать функцию PerimN(*P*, *N*) вещественного типа, находящую периметр *N*-угольника, вершины которого (в порядке их обхода) передаются в массиве *P* размера *N* (> 2) с элементами типа TPoint. С помощью этой функции найти периметры трех многоугольников, если дано число их сторон и координаты их вершин.
3. Используя типы TPoint, TTriangle и функцию Area, описать функцию AreaN(*P*, *N*) вещественного типа, находящую площадь выпуклого *N*-угольника, вершины которого (в порядке их обхода) передаются в массиве *P* размера *N* (> 2) с элементами типа TPoint. С помощью этой функции найти площади трех многоугольников, если дано число их сторон и координаты их вершин.

# **Литература**

1. Абрамов С. А., Зима Е. В. Начала информатики. – М.: Наука, 1989. – 256 с.
2. Абрамян М. Э. Практикум по программированию на языке Паскаль. 4-е изд. — Ростов-на-Дону: Изд-во «ЦВВР», 2004. — 187 с.
3. Абрамян М. Э., Михалкович С. С. Основы программирования на языке Паскаль: Скалярные типы данных, управляющие операторы, процедуры и функции. — Ростов-на-Дону: Изд-во «ЦВВР», 2004. — 198 с.
4. Абрамян М.Э. 1000 задач по программированию. Часть I (Скалярные типы данных, управляющие операторы, процедуры и функции). - Ростов-на-Дону, 2004. – 43 С.
5. Абрамян М.Э. 1000 задач по программированию. Часть II (Минимумы и максимумы, одномерные и двумерные массивы, символы и строки, двоичные файлы). - Ростов-на-Дону, 2004. – 42 С.
6. Абрамян М.Э. 1000 задач по программированию. Часть III (Текстовые файлы, составные типы данных в процедурах и функциях, рекурсия, указатели и динамические структуры). - Ростов-на-Дону, 2004. – 43 С.
7. Арабов М.К., Нуров И.Дж. Основы программирования на языке PascalABC.NET. Учебно-методическое пособие. – Душанбе: РТСУ; 2018. - 280 с.
8. Арабов М.К. Основы программирования на С++. - Душанбе. РТСУ, 2018. - 400с.
9. Арабов М.К., Каландаров Х.Х. Программирование на языке Python: учебное пособие для студентов- бакалавров с 2 курса направления подготовки «Прикладная информатика» / М.К. Арабов, Х.Х. Каландаров. Душанбе: РТСУ, 2021. - 261 с.
10. Арабов М.К. Современные технологии Java: учебное пособие для студентов- бакалавров с 1- 4 курсов и магистрантов первого по второго года обучения направления подготовки «Прикладная информатика» а также школьников старших классов / М.К. Арабов. Душанбе: РТСУ, 2020. - 293 с.
11. Габасов Р., Кириллова Ф. М. Основы динамического программирования. — Мн.: Изд-во БГУ, 1975. — 262 с.
12. Герберт Шилдт. C: полное руководство, классическое издание.— М.: Вильямс, 2010. — 704 с.
13. Герберт Шилдт. Полный справочник по C++ . — М.: Вильямс, 2011. — 800 с.
14. Голуб А. И. С и С++. Правила программирования. - М: БИНОМ, 1996 г. - 272 с.
15. Гукин Д. Язык программирования Си для «чайников».— М.: Диалектика, 2006. — 352 с.
16. Дейтел Х., Дейтел П. Как программировать на С++.— М.: БИНОМ, 1998 - 1024 с.
17. М. Арабов, М. Халилова. Сборник примеров и задач по программированию. - Душанбе. РТСУ, 2018. - 138 с.
18. Муллошараф Арабов. Основы программирования на языке PascalABC.NET. - Душанбе. РТСУ - 2018. - 302с.
19. Шилдт Г. C# 4.0: полное руководство.: Пер. с англ. — М.: ООО "И.Д. Вильямс", 2011. — 1056 с.
20. Шилдт Г. Полный справочник по C++ .— М.: Вильямс, 2011.- 800 с.
21. Шилдт Г. Самоучитель C++: Пер. с англ. - Санкт-Петербург: BHV-Санкт-Петербург, 1998. – 620 с.
22. [https://github.com/](https://github.com/swaroopch/byte_of_python)
23. [https://pythonworld.ru/](https://pythonworld.ru/samouchitel-python)
24. <http://golang-book.ru/>
25. <https://proglib.io/p/30-golang-books/>
26. <http://cppstudio.com>
27. https://metanit.com/

1. *Мирча Элиаде (рум. Mircea Eliade; 13 марта*[*1907*](https://ru.wikipedia.org/wiki/1907_%D0%B3%D0%BE%D0%B4)*, Бухарест, Королевство Румыния —*[*22 апреля*](https://ru.wikipedia.org/wiki/22_%D0%B0%D0%BF%D1%80%D0%B5%D0%BB%D1%8F)[*1986*](https://ru.wikipedia.org/wiki/1986_%D0%B3%D0%BE%D0%B4)*, Чикаго, США) —*[*румынский*](https://ru.wikipedia.org/wiki/%D0%A0%D1%83%D0%BC%D1%8B%D0%BD%D0%B8%D1%8F)*,*[*французский*](https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%B0%D0%BD%D1%86%D1%83%D0%B7%D1%81%D0%BA%D0%B8%D0%B9)*и американский философ, философ культуры,* [*религиовед*](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D0%B8%D0%B3%D0%B8%D0%BE%D0%B2%D0%B5%D0%B4)*, историк религий, этнограф и писатель.* [↑](#footnote-ref-0)