

*УТВЕРЖДАЮ:*

Начальник управления образования  
администрации г. Кемерово

---

Н.Ю. Дашковская

« \_\_\_\_\_ » \_\_\_\_\_ 2024 г.

**Положение  
о проведении открытого городского турнира по  
программированию среди учащихся**

Кемерово, 2024

## **I. Общие положения**

1.1. Городской турнир по программированию (далее Турнир) проводится с целью создания среды для обмена опытом, демонстрации и совершенствования способностей обучающихся по программированию, развития познавательного интереса к изучению возможностей языков программирования, использования информационно – коммуникационных технологий.

1.2. Задачи Турнира:

- популяризировать программирование среди обучающихся;
- способствовать повышению уровня алгоритмического мышления обучающихся;
- способствовать выявлению одарённых и талантливых школьников, их дальнейшего интеллектуального развития и профессиональной ориентации.

## **II. Руководство и организация проведения Турнира**

2.1. Общее руководство Турниром осуществляет управление образования администрации г. Кемерово.

2.2. Подготовку и проведение Турнира осуществляет Центр цифрового образования IT – Куб г. Кемерово (на базе МБОУДО «Центр дополнительного образования детей и юношества Кировского района»), консультантами Турнира являются учителя информатики ОУ г. Кемерово, преподаватели АНО ПО «Колледж предпринимательских и цифровых технологий»

2.3. Деятельность по подготовке и проведению Турнира координирует Оргкомитет в составе:

- Гусева Анастасия Камильевна, главный специалист отдела общего и дополнительного образования управления образования администрации г. Кемерово;
- Чепурина Татьяна Александровна, заведующий отделом по работе с одарёнными детьми МБОУ ДПО «НМЦ»
- Вяткина Таисия Викторовна, директор МБОУДО «ЦДОДиЮ Кировского района»
- Швед Яна Юрьевна, зам директора по УВР МБОУДО «ЦДОДиЮ Кировского района»
- Агеева Яна Александровна, заведующий отделом технического творчества МБОУДО «ЦДОДиЮ Кировского района».

Оргкомитет Конкурса формирует жюри Турнира из числа педагогических работников Центра цифрового образования IT – Куб, учителей информатики школ г. Кемерово, преподавателей и студентов Колледжа предпринимательских и цифровых технологий, а также технологических партнеров Центра цифрового образования IT-Куб г. Кемерово.

### III. Сроки и место проведения

3.1 Турнир проводится на базе МБОУДО ЦДОДиЮ Кировского района (Центр цифрового образования IT – Куб г. Кемерово) по адресу ул. Гурьевская 14 А.

*Дата проведения: 28 и 29 марта 2024 года (28 марта - участники стартового уровня, 29 марта – участники базового и продвинутого уровней)*

*Время регистрации участников: 9.30 – 10.00 - участники 1 потока; 12.00 – 12.30 участники 2 потока.*

*Время проведения Турнира: 10.00 – 12.00 участники первого потока, 12.30 – 14.30*

График проведения Турнира и вся дополнительная информация будет опубликована в телеграмм-канале <https://t.me/writeKode> (телеграмм-канал для руководителей и участников)

*Место проведения:* г. Кемерово, ул. Гурьевская, 14А, МБОУДО ЦДОДиЮ Кировского района, тел. 8 (3842) 25-64-73, e-mail: [crt\\_kir@mail.ru](mailto:crt_kir@mail.ru)

Заявки на участие в Турнире принимаются в срок до **25 марта 2024 года** по ссылке <https://docs.google.com/spreadsheets/d/1TA27ZlmsZMG9uZsojdFNmkeWuFSdDFbdKVZBmt23CZo/edit?usp=sharing>

При регистрации в день проведения Турнира участнику необходимо предоставить Согласие на обработку персональных данных (Приложение 1)

### IV. Порядок и условия проведения Турнира.

4.1 Участие в Турнире – индивидуальное.

4.2. Участие в Турнире принимают учащиеся образовательных организаций в возрасте **от 7 до 17 лет.**

4.3. Возрастные группы участников: 7 – 9 лет, 10 – 11 лет, 12 – 13 лет, 14 – 17 лет

### V. Содержание и порядок проведения Турнира

5.1. Задания Турнира делятся на 3 уровня сложности:

- стартовый уровень (для возрастных категории 7 – 9 лет, 10 – 11 лет, 12 – 13 лет)
- базовый уровень (14 – 17 лет)
- продвинутый уровень (14 – 17 лет)

5.2. Участник возрастной категории 14 – 17 лет вправе сам выбирать уровень сложности заданий (*базовый /продвинутый*)

5.3. Задания стартового уровня определяют навыки алгоритмического мышления учащегося.

5.3.1. Задания **стартового уровня** представляют собой решение алгоритмических задач на платформе ПиктоМир. Участнику необходимо решить максимальное количество задач за **30 минут** отведенного времени.

5.4. Задания базового и продвинутого уровня определяют уровень знаний участника Турнира по следующим критериям:

- *линейные алгоритмы*

- условные операторы
- одномерные массивы
- работа в среде программирования КуМир

5.4.1 Задания **базового и продвинутого уровней** представляют собой кейсы ситуационных задач, решить которые возможно на одном из предложенных языков программирования: Python, JAVA, Paskal (язык программирования участник выбирает самостоятельно).

5.5. Задания выполняются на ноутбуках, с заранее установленной средой программирования. Решения задач Турнира сохраняются в указанную папку на «рабочем столе» в виде скриншотов с экрана.

5.6. С примерными заданиями Турнира можно ознакомиться в Приложении № 2

## **VI. Подведение итогов Турнира и награждение.**

6.1 Победители и призеры Турнира определяются по количеству набранных баллов. Победители и призёры Турнира награждаются грамотами управления образования администрации г. Кемерово.

6.2. Победители и призеры определяются в каждой возрастной группе участников (в соответствии с п 4.3. настоящего Положения)

6.3 Наставники, подготовившие Победителей и призеров, награждаются благодарственными письмами Центра цифрового образования ИТ - Куб. Все участники Турнира получают сертификат участника в электронном виде на почту, указанную при регистрации.

6.4 Результаты Турнира будут опубликованы на официальной странице ЦДОДиЮ Кировского района Вконтакте [https://vk.com/crt\\_kir](https://vk.com/crt_kir) не позднее, чем через 7 дней после проведения.

## **VIII. Финансирование**

8.1. Необходимое оборудование для проведения Турнира предоставляет МБОУДО ЦДОДиЮ Кировского района (Центр цифрового образования детей ИТ – Куб г. Кемерово)

8.2. Расходы, связанные с участием в Турнире (проезд к месту проведения) несут командирующие организации.

8.3. Наличие сменной обуви для всех участников – обязательно.

✉ 650033, г. Кемерово, ул. Гурьевская, 14 А, МБОУДО ЦДОДиЮ Кировского района (Центр цифрового образования детей ИТ – Куб)

☎ (3842) 25-64-74 Швед Яна Юрьевна 89089445325

E-mail: [crt\\_kir@mail.ru](mailto:crt_kir@mail.ru)

СОГЛАСИЕ ЗАКОННОГО ПРЕДСТАВИТЕЛЯ НА ОБРАБОТКУ ПЕРСОНАЛЬНЫХ ДАННЫХ НЕСОВЕРШЕННОЛЕТНЕГО

Я, \_\_\_\_\_ (ФИО),

проживающий по адресу \_\_\_\_\_,

являюсь законным представителем несовершеннолетнего

\_\_\_\_\_ (ФИО) на основании ст. 64 п. 1 Семейного

кодекса РФ. Настоящим даю свое согласие на обработку муниципальным органом управления образования персональных данных моего несовершеннолетнего ребенка

\_\_\_\_\_, относящихся исключительно к

перечисленным ниже категориям персональных данных: основное место учебы

(образовательное учреждение, класс) другие образовательные учреждения,

посещаемые ребенком (ОУДОД и др.) Я даю согласие на использование

персональных данных моего ребенка исключительно в следующих целях: - участие

в городской Олимпиаде по программированию - ведение статистики. Настоящее

согласие предоставляется на осуществление сотрудниками муниципального органа

управления образования следующих действий в отношении персональных данных

ребенка: сбор, систематизация, накопление, хранение, уточнение (обновление,

изменение), публикацию (в том числе, в сети Интернет на веб-сайтах

<http://kemedu.ucoz.ru>), использование (только в указанных выше целях),

обезличивание, блокирование (не включает возможность ограничения моего

доступа к персональным данным ребенка), уничтожение. Я не даю согласия на

какое-либо распространение персональных данных ребенка, в том числе на передачу

персональных данных ребенка каким-либо третьим лицам, включая физические и

юридические лица и учреждения, государственные органы и органы местного

самоуправления для осуществления обработки персональных данных. Обработку

персональных данных ребенка для любых иных целей и любым иным способом,

включая распространение и передачу каким-либо третьим лицам, я запрещаю. Она

может быть возможна только с моего особого письменного согласия в каждом

отдельном случае. Данное Согласие действует до достижения целей обработки

персональных или до отзыва данного Согласия. Данное Согласие может быть

отозвано в любой момент по моему письменному заявлению. Я подтверждаю, что,

давая настоящее согласие, я действую по своей воле и в интересах ребенка,

законным представителем которого являюсь.

Дата: \_\_\_\_\_ г. Подпись: \_\_\_\_\_

**Базовый уровень**  
**Задачи на программирование**  
**Линейные алгоритмы**

1. Скорость первого автомобиля  $V_1$  км/ч, второго —  $V_2$  км/ч, расстояние между ними  $S$  км. Определить расстояние между ними через  $T$  часов, если автомобили первоначально движутся навстречу друг другу. Данное расстояние равно модулю разности начального расстояния и общего пути, проделанного автомобилями; общий путь = время · суммарная скорость.

**Python**

Решение:

```
import random
V1,V2 = [random.randrange(50,70) for i in range(2)]
S = random.randrange(0,20)
T = random.randrange(1,6)
total_speed = V1 + V2
total_path = total_speed * T
D = S + total_path
print("1st vehicle speed:",V1)
print("2nd vehicle speed:",V2)
print("Time:",T)
print("Total speed:",total_speed)
print("Initial Distance between vehicles:",S)
print("Final Distance between vehicles:",D)
```

**Pascal**

Решение:

```
program Begin36;
var
  V1,V2,S,T,SResult: Real;
begin
  Write('Введите скорость первого автомобиля: ');
  Readln(V1);
  Write('Введите скорость второго автомобиля: ');
  Readln(V2);
  Write('Введите расстояние между автомобилями : ');
  Readln(S);
  Write('Введите прошедшее время: ');
  Readln(T);
  SResult:=S+V1*t+V2*T;
  Writeln('Между автомабиями: ',SResult);
end.
```

**Java**

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
```

```
System.out.println("Введите скорость первого автомобиля (км/ч): ");
double V1 = scan.nextDouble();
```

```
System.out.println("Введите скорость второго автомобиля (км/ч): ");
double V2 = scan.nextDouble();
```

```
System.out.println("Введите время, через которое автомобили должны
встретиться (ч): ");
double T = scan.nextDouble();
```

```
System.out.println("Введите начальное расстояние ");
double S = scan.nextDouble();
```

```
double SResult = Math.abs(S-(V1*T+V2*T));
```

```
System.out.println("Через " + T + " часов расстояние между автомобилями
будет равно " + SResult);
    }
}
```

### Условные операторы

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

#### Python

Решение:

```
import random
Y = random.randint(1,3000)
#Y = 1901
print("Год:", Y)
s = int((Y-1)/100)+1
print("Столетие:", s)
```

#### Pascal

Решение:

```
program Integer30;
var
    Year, Century: Integer;
begin
    Write('Введите номер года: ');
    Readln(Year);
    Century:=((Year-1) div 100)+1;
    Writeln('Это год ', Century, ' столетия. ');
End.
```

#### Java

```
import java.util.Scanner;
public class Century {
```

```

public static void main(String[] args) {
    Scanner scan = new Scanner(System.in);
    int year = scan.nextInt();
    int Century = (year/100)+1;
    System.out.println(Century + " век");
}
}

```

### Одномерные массивы

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

#### Python

Решение:

```

import random
N = random.randrange(1,10)
print("N = ", N)
a1 = []
for i in range(N):
    a1.append(i*2+1)
print(a1)
a2 = [i*2+1 for i in range(N)]
print(a2)

```

#### Pascal

Решение:

```

program Array1;
var
    a:array[1..10] of integer;
    N,i:Integer;
begin
    Write('N:');
    Readln(N);
    for i:=1 to N do a[i]:=i*2-1;
    for i:=1 to N do writeln(i, ' : ',a[i]);
end.

```

#### Java

```

import java.util.Scanner;
public class task3 {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("N:");
        int n = in.nextInt();
        int[] a = new int[n];

        for (int i = 0; i < n; i++)
            a[i] = 1 + i * 2;
        for (int i = 0; i < n; ++i)
            System.out.println("a[" + i + "]=" + a[i]);
        in.close();
    }
}

```

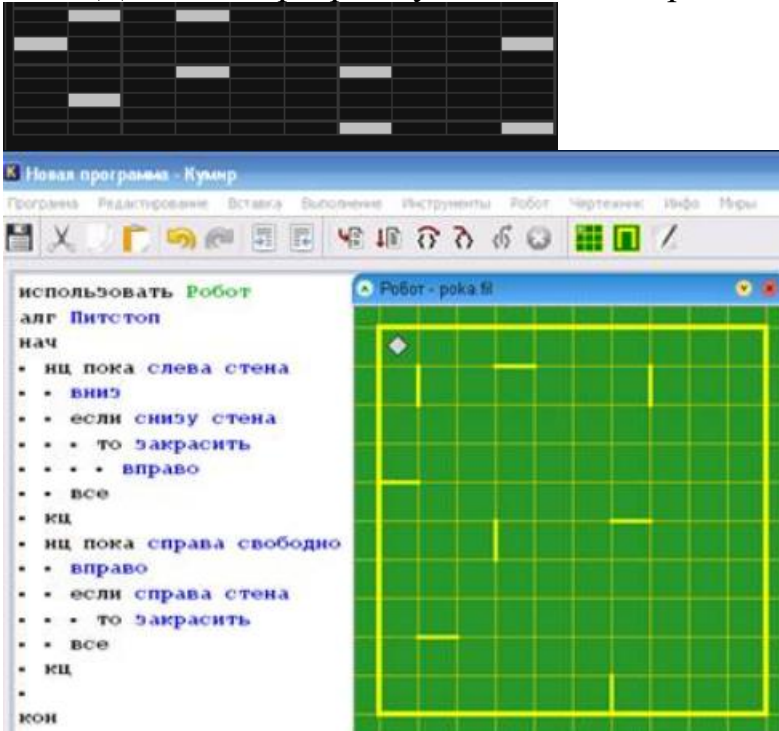


}

## Среда программирования КуМир

**Задание № 2: ПИТСТОП. (25 баллов – программа написана полностью, 15 баллов – программа написана частично)**

1. Создать предлагаемую обстановку;
2. Сохранить как PitStop;
3. Написать часть программы;
4. Дописать программу так, чтобы закрасить клетки, как показано в образце.



### Продвинутый уровень Операторы цикла

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

### Python

Решение:

```
import random
A,B = [random.randrange(1, 21) for i in range(0, 2)]
C = random.randint(1,min(A,B))
print("A = {0}, B = {1}, C = {2}".format(A,B,C))
a = int(A/C)
b = int(B/C)
print("Площадь прямоугольника:",A*B)
print("Площадь квадрата:",C*C)
print("Количество квадратов на прямоугольнике:",a*b)
```

### Pascal

Решение:

```

program While30;
var
  A,B,Btemp,C:Real;
  K:integer;
begin
  Write('Введите A: ');
  Readln(A);
  Write('Введите B: ');
  Readln(B);
  Write('Введите C: ');
  Readln(C);
  K:=0;

  while (A-C>=0)do
  begin
    A:=A-C;
    Btemp:=B;
    while (Btemp-C>=0) do
    begin
      Btemp:=Btemp-C;
      inc(K,1);
    end;
  end;
  Writeln(k);
end.

```

## Java

```

import java.util.Scanner;
public class AdvancedFirst {
  public static void main(String[] args) {
    Scanner scan = new Scanner(System.in);
    System.out.println("Введите A: ");
    double A = scan.nextDouble();
    System.out.println("Введите B: ");
    double B = scan.nextDouble();
    System.out.println("Введите C: ");
    double C = scan.nextDouble();
    int K = 0;
    while (A-C>=0){
      A= A-C;
      double Btemp = B;
      while (Btemp>=0){
        Btemp = Btemp - C;
        ++K;
      }
    }
    System.out.println("K: " + K );
  }
}

```

## Одномерные массивы

1. Дан массив  $A$  размера  $N$ . Не изменяя данный массив, вывести номера его элементов в том порядке, в котором соответствующие им элементы образуют возрастающую последовательность. Использовать метод «пузырьковой» сортировки, модифицировав его следующим образом: создать вспомогательный целочисленный массив номеров  $I$ , заполнив его числами от 1 до  $N$ ; просматривать массив  $A$ , сравнивая пары элементов массива  $A$  с номерами  $I_1$  и  $I_2$ ,  $I_2$  и  $I_3$ , ... и меняя местами соответствующие элементы массива  $I$ , если левый элемент пары больше правого. Повторив описанную процедуру просмотра  $N-1$  раз, получим в массиве  $I$  требуемую последовательность номеров.

### Python

Решение:

```
import random
N = random.randrange(2,10)
#a = [2*(i+1) for i in range(N)]
A = [random.randrange(0,10) for i in range(N)]
I = [i for i in range(N)]
print("N = ", N)
print("Array A:")
print(A)
print("Array I:")
print(I)
for i in range(1,N) :
    for j in range(0,N-1) :
        if A[I[j]] > A[I[j+1]] :
            I[j], I[j+1] = I[j+1], I[j]
print("Sorted Array:")
for i in range(0,N) :
    print(A[I[i]],end=" ")
##print()
##print("Array A:")
##print(A)
##print("Array I:")
##print(I)
```

### Pascal

Решение:

```
program Array115;
var
  a : array [1..6] of real;
  i: array [1..6] of integer;
  N,N2,k,k2,k3: integer;

begin
  Write('N: ');
  Readln(N);

  Writeln('A: ');
```

```

for k:=1 to N do
begin
write(k, ':');
readln(a[k]);
i[k]:=k;
end;
N2:=N;
for k:=1 to N-1 do
begin
k2:=1;
N2:=N2-1;
while (k2<=N2) do begin if a[i[k2]] > a[i[k2]+1] then
begin
i[k2]:=i[k2]+i[k2+1];
i[k2+1]:=i[k2]-i[k2+1];
i[k2]:=i[k2]-i[k2+1];
end;
inc(k2);
end;
for k3:=1 to N do writeln(i[k3]);
readln;
end;
end.

```

## Java

```

import java.util.Scanner;
public class AdvancedSecond {
public static void main(String[] args) {
int[] a = new int[20];
int[] i = new int[20];
Scanner scan = new Scanner(System.in);
System.out.println("N: ");
int n = scan.nextInt();
int i4;
for (i4 = 0; i4 < n; ++i4) {
System.out.println("a[%i] : " + i4 + 1);
i[i4] = i4;
}
int i2, i3;
for (i4 = 0; i4 < n; ++i4) {
for (i2 = 0; i2 <= n - i4 - 2; ++i2) {
if (a[i[i2]] > a[i[i2] + 1]) {
i[i2] += i[i2 + 1];
i[i2 + 1] = i[i2] - i[i2 + 1];
i[i2] -= i[i2 + 1];
}
}
}
for (i3 = 0; i3 < n; ++i3){
i3 = i3 + 1;
i[i3] = i[i3] + 1;
}
}
}

```

```

        System.out.println(i3);
        System.out.println(i[i3]);
    }
}
}}
```

## Линейные алгоритмы

1. Найти корни квадратного уравнения  $A \cdot x^2 + B \cdot x + C = 0$ , заданного своими коэффициентами  $A$ ,  $B$ ,  $C$  (коэффициент  $A$  не равен 0), если известно, что дискриминант уравнения положителен. Вывести вначале меньший, а затем больший из найденных корней. Корни квадратного уравнения находятся по формуле  $x_{1, 2} = (-B \pm \sqrt{D}) / (2 \cdot A)$ , где  $D$  — дискриминант, равный  $B^2 - 4 \cdot A \cdot C$ .

### Python

Решение:

```

import random
import math
r = list(range(-10,0)) + list(range(1,11))
A = random.choice(r)
C = random.randrange(-10,11)
B1 = math.ceil(math.sqrt(4*A*C))

B = random.randrange(B1,B1+10)
coef = random.choice([-1,1])

B = coef * B

print("Ax^2 + Bx + C = 0")
print("A = ",A)
print("B = ",B)
print("C = ",C)

D = B*B - 4*A*C
x1 = (-B - math.sqrt(D))/(2*A)
x2 = (-B + math.sqrt(D))/(2*A)
s = sorted([x1,x2])
print("D = ",D)
print(s)
print("Check x1:",A*x1*x1+B*x1+C)
print("Check x2:",A*x2*x2+B*x2+C)
```

### Pascal

Решение:

```

program Begin39;
var
  A,B,C,D,x1,x2: Real;
begin
  Write('Введите коэффициент A: ');
  Readln(A);
  Write('Введите коэффициент B: ');
```

```

Readln(B);
Write('Введите коэффициент C: ');
Readln(C);
D:=Sqr(b)-4*A*C;
x1:=((-1)*B+sqrt(D))/(2*A);
x2:=((-1)*B-sqrt(D))/(2*A);
if x2<x1 then
begin
  Writeln('Меньший корень равен : ',X2);
  Writeln('Большой корень равен : ',X1);
end
else
begin
  Writeln('Меньший корень равен : ',X1);
  Writeln('Большой корень равен : ',X2);
end;
end.

```

### Java

```

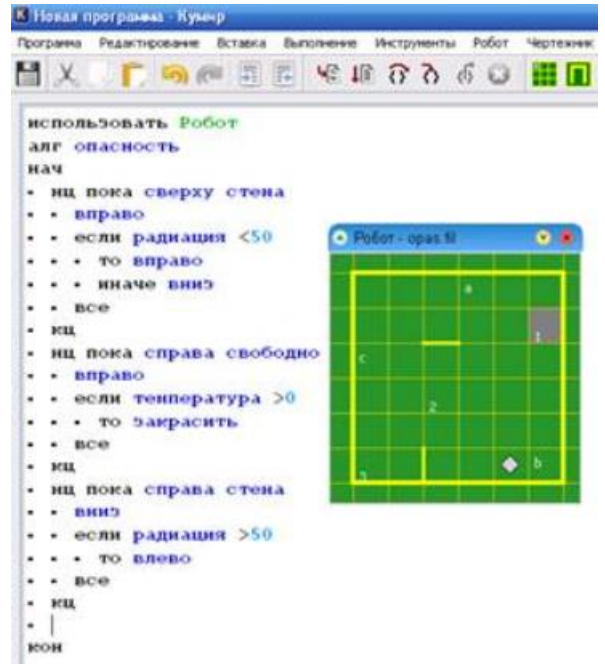
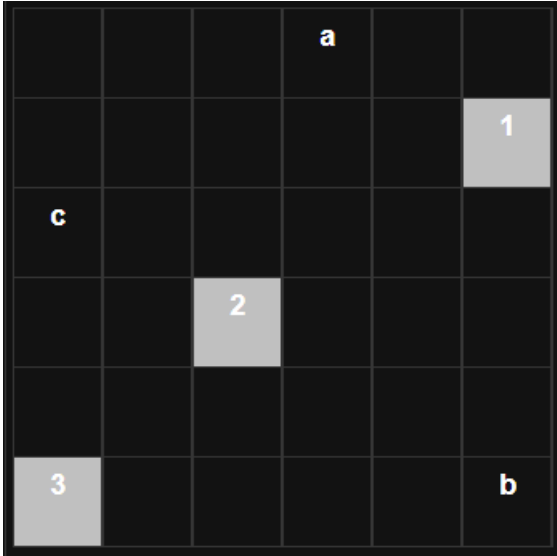
import java.util.Scanner;
public class ThirdAdvanced {
  public static void main(String[] args) {
    Scanner scan = new Scanner(System.in);
    System.out.println("Введите коэффициент A: ");
    double A = scan.nextDouble();
    System.out.println("Введите коэффициент B: ");
    double B = scan.nextDouble();
    System.out.println("Введите коэффициент C: ");
    double C = scan.nextDouble();
    double D = Math.pow(B,2)-4*A*C;
    double x1 = ((-1)*B+Math.sqrt(D))/(2*A);
    double x2 = ((-1)*B-Math.sqrt(D))/(2*A);
    if(x2<x1){
      System.out.println("Меньший корень равен: " + x2);
      System.out.println("Меньший корень равен: " + x1);
    }
    else{
      System.out.println("Меньший корень равен: " + x1);
      System.out.println("Меньший корень равен: " + x2);
    }
  }
}

```

# Среда программирования КуМир

## Задание № 1: ОПАСНОСТЬ.

1. Создать предлагаемую обстановку (робот в левом верхнем углу, клетка с радиацией – метка «буква», клетка с температурой – метка «цифра»);
2. Сохранить как Opanost;
3. Написать часть программы;
4. Дописать программу так, чтобы робот реагировал (смещался) на клетки с радиацией (клетка с буквой) и закрашивал клетки с температурой (клетка с цифрой):  
а – 1 – b – 2 – с–3.



Справка: цикл пока создается структурами  
нц пока слева (справа, снизу, сверху) стена влево (вправо, вниз, вверх) кц