

Решения

Задач заключительного тура олимпиадной части конкурса. 11 класс

1. (математика) Вероятность обнаружения блохи Кузи в вершинах A, B, C и D квадрата $ABCD$ равны p_1, p_2, p_3 и p_4 соответственно ($p_1 + p_2 + p_3 + p_4 = 1$). Блоха прыгает в ближайшую к ней вершину квадрата, причем в какую из двух – совершенно случайно и равновозможно. Найти вероятность обнаружить Кузю в вершине B после ее 2019 прыжка.

Решение. Пусть $P_A(n), P_B(n), P_C(n), P_D(n)$ – вероятности обнаружить блоху в вершинах A, B, C и D после n прыжков. Тогда

$$P_A(1) = \frac{1}{2}(p_2 + p_4) = P_C(1), \quad P_B(1) = \frac{1}{2}(p_1 + p_3) = P_D(1)$$

На рис 1 изображены вероятности событий после первого прыжка.

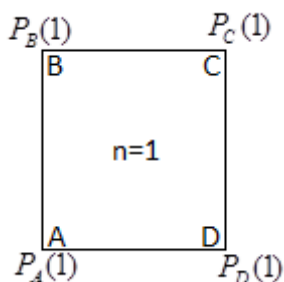


Рис 1

После второго прыжка

$$P_A(2) = \frac{1}{2}P_B(1) + \frac{1}{2}P_D(1) = \frac{1}{2}(p_1 + p_3) = P_B(1) = P_D(1) = P_C(2)$$

$$P_B(2) = \frac{1}{2}P_A(1) + \frac{1}{2}P_C(1) = \frac{1}{2}(p_2 + p_4) = P_A(1) = P_C(1) = P_D(2)$$

На рис 2 изображены вероятности событий после второго прыжка.

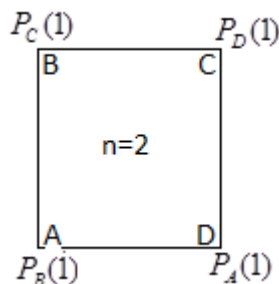


Рис 2

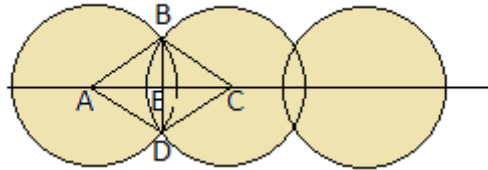
Картинка при $n=2$ может быть получена из картинке при $n=1$ поворотом на 90° против часовой стрелки, а поворот ее на 180° оставляет распределение вероятностей неизменным. Поэтому для

всех нечетных n распределение вероятностей совпадает с изображенным на рис 1, а для четных n – на рис 2. Следовательно, для $n = 2019$ вероятность $P_B(2019) = P_B(1) = \frac{1}{2}(p_1 + p_3) = 0,2$.

2. (математика) Тринадцать кругов одинакового радиуса $r = 2\sqrt{2-\sqrt{3}}$ вырезаны из цветной бумаги и разложены на плоскости так, что их центры лежат на одной прямой. Расстояние между центрами соседних кругов одинаковое и равно 2. Найти периметр фигуры на плоскости, образованной этими кругами.

Ответ: $P = 44\pi\sqrt{2-\sqrt{3}}$

Решение:



Круги пересекаются поскольку $r > 1$. На рисунке изображены три из тринадцати кругов.

$$AC = 2, AB = BC = r = 2\sqrt{2-\sqrt{3}}, \angle BAE = \alpha, \cos \alpha = \frac{AE}{AB} = \frac{1}{2\sqrt{2-\sqrt{3}}} = \frac{\sqrt{2+\sqrt{3}}}{2}$$

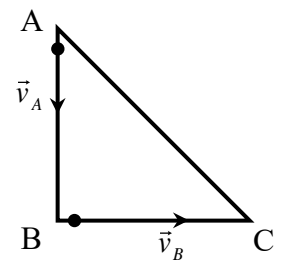
Вычислим $\cos 2\alpha$:

$$\sin^2 \alpha = 1 - \cos^2 \alpha = 1 - \frac{2+\sqrt{3}}{4} = \frac{2-\sqrt{3}}{4} \rightarrow \cos 2\alpha = \cos^2 \alpha - \sin^2 \alpha = \frac{\sqrt{3}}{2} \rightarrow 2\alpha = 30^\circ$$

$$\text{Длина дуги } BD: L_{BD} = \frac{2\pi r}{12} = \frac{\pi}{6}\sqrt{2-\sqrt{3}}$$

$$\text{Периметр фигуры } P = 13 \cdot 2\pi r - 24L_{BD} = 26\pi r - 24 \cdot \frac{\pi r}{6} = 22\pi r = 44\pi\sqrt{2-\sqrt{3}}.$$

3. (физика) Из проволоки сделан прямоугольный треугольник с катетами длиной $l = 1$ м. Из вершин A и B по катетам одновременно начинают ползти два жука со скоростями $v_A = 5$ см/с и $v_B = 2v_A = 10$ см/с (см. рисунок). Через какое минимальное время после начала движения расстояние между жуками достигнет минимума?



Решение. Пусть после начала движения жуков прошло время t . Тогда жук

A пройдет расстояние $v_A t$, жук B – расстояние $v_B t = 2v_A t$. Следовательно, расстояние S между жуками в этот момент будет равно

$$S = \sqrt{(l - v_A t)^2 + 4v_A^2 t^2} \quad (*)$$

Найдем минимум величины S как функции времени. Для этого продифференцируем функцию $S(t)$, приравняем производную к нулю и решим полученное уравнение. Причем технически удобнее дифференцировать не S , а S^2 , поскольку выражения будут гораздо проще, а максимума и величина S , и величина S^2 достигают при одном значении аргумента. Итак

$$(S^2)' = -2(l - v_A t)v_A + 8v_A^2 t \quad (**)$$

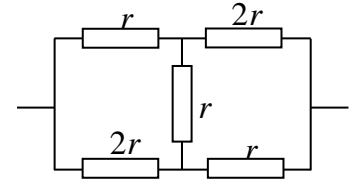
Приравнявая производную (***) к нулю, получаем уравнение относительно времени, прошедшего с начала движения до того момента, когда расстояние между жуками будет наименьшим

$$l - v_A t = 4v_A t$$

Откуда

$$t = \frac{l}{5v_A} = 4 \text{ с}$$

4. (физика) Найти общее сопротивление электрической цепи, схема которой представлена на рисунке. Значения всех сопротивлений приведены на рисунке. $r = 1$ Ом.



Решение. Пусть в рассматриваемый участок втекает электрический ток I , который делится на первом разветвлении на токи I_1 (в нижнем колене цепи), и $I - I_1$ (в верхнем колене; см. рисунок). Благодаря симметрии участка, после среднего сопротивления r токи поменяются местами - $I - I_1$ в нижнем колене, и I_1 - в верхнем. Поэтому через среднее сопротивление r течет электрический ток $2I_1 - I$ (от нижнего узла к верхнему) и разность потенциалов между этими узлами можно найти по закону Ома для участка цепи

$$\varphi_2 - \varphi_1 = r(2I_1 - I) \quad (*)$$

С другой стороны, разность потенциалов между этими узлами можно найти по закону Ома для верхнего участка с сопротивлением r и нижнего участка с сопротивлением $2r$

$$\varphi_2 - \varphi_1 = (\varphi_0 - \varphi_1) - (\varphi_0 - \varphi_2) = r(I - I_1) - 2rI_1 = r(I - 3I_1) \quad (**)$$

Приравнявая формулы (*) и (**), получим

$$2I_1 - I = I - 3I_1$$

Отсюда находим ток $I_1 = \frac{2}{5}I$ и разность потенциалов между левым и правым узлами цепи

$$\varphi_0 - \varphi_3 = \frac{3}{5}Ir + \frac{2}{5}I2r = \frac{7}{5}Ir.$$

Следовательно,

$$R_{\text{об}} = \frac{7}{5}r = 1,4 \text{ Ом.}$$

5. (информатика) В квадратной матрице $N \times N$ все клетки заполнены числами от 1 до 5. Связной компонентой в матрице будем называть такое множество ячеек матрицы, которые заполнены одним и тем же числом, и между любыми ячейками множества можно построить путь. Соединяться в путь могут только соседние по горизонтали или вертикали ячейки.

Иными словами, связная компонента матрицы – это связная фигура, ячейки которой заполнены одним и тем же числом.

Для заданной матрицы требуется найти связную компоненту, состоящую из наибольшего количества ячеек.

1	2	5	3	5	2
1	2	5	5	5	2
1	2	2	2	4	5
1	2	5	2	5	5
1	3	5	2	5	5
1	3	5	5	4	4

В первой строке входных данных содержится целое число N , такое что $1 \leq N \leq 10$. В следующих N строках находятся N целых чисел, разделенных пробелами, каждое число лежит в пределах от 1 до 5. В качестве ответа нужно вывести единственное число – количество ячеек, из которого состоит наибольшая по величине связная компонента матрицы.

Примеры входных данных и результатов работы программы:

<i>Пример входных данных</i>	<i>Пример результата</i>
6 1 2 5 3 5 2 1 2 5 5 5 2 1 2 2 2 4 5 1 2 5 2 5 5 1 3 5 2 5 5 1 3 5 5 4 4	8

Комментарии к задаче по информатике:

Решением задачи является код программы, написанный на любом традиционном языке программирования, с указанием этого языка. В случае невозможности написать код на традиционном языке программирования, в качестве частичного решения может быть принят правильный алгоритм программы, оформленный в виде блок-схемы или псевдокода.

Программа должна читать входные данные из стандартного потока ввода (так, как будто эти данные вводятся с клавиатуры) и выводить результаты в стандартный поток вывода (так, как будто эти данные печатаются на экран). Программа должна корректно работать на входных данных, описанных в условиях задачи. Корректность входных данных гарантируется, проверять её дополнительно внутри программы не требуется (если в условии задачи сказано, что на вход подается целое число от 0 до 1000, не нужно дополнительно проверять, что введена, например, текстовая строка, или число вне этого диапазона).

Программа должна в результате работы выводить только тот ответ, который требуется по условию (может сопровождаться кратким текстовым оформлением или без него). Любой другой вывод результатов в процессе работы программы будет считаться ошибкой. Для лучшего понимания условий задачи и форматов входных и выходных данных, задача сопровождается несколькими примерами корректных входных данных и правильного результата работы, приведенных в разделе «*примеры входных данных и результатов работы программы*».

Написанная программа должна работать *эффективно*, то есть вычислять правильный ответ, по возможности за наименьшее время. Программы, написанные *существенно неэффективно*, то есть затрачивающие существенно больше времени, чем эффективные решения, будут считаться неполным решением.

Код программы должен быть написан разборчиво, аккуратно, сопровожден отступами и разумным количеством комментариев в коде программы. Допускается несколько несущественных синтаксических ошибок в коде, при условии, что они не влияют на общую возможность чтения и понимания кода. Неаккуратно написанный (нечитаемый) код программы и/или большое количество синтаксических ошибок могут привести к снижению общей оценки за задачу.

Пример оформления решения (кода программы):

```
{pascal} //указание языка, на котором написана программа
programMyProg;
var: ...
begin
    {считываем входные данные} //комментарии в теле программы
readln(a,b);
    ...
    {основное тело программы}
    ...
    {выводим ответ}
writeln('Искомое число: ',x);
end.
```

Решение.

```
program prog11;
var n,i,j,k,l:integer;
    a,b:array[0..11,0..11] of integer;
    m,mc,mmax:integer;
begin
    {инициализируем массив}
for i:=0 to 11 do
    for j:=0 to 11 do
```

```

a[i,j]:=0;
{читаем входные данные}
readln(N);
for i:=1 to N do
begin
  for j:=1 to N do
    read(a[i,j]);
  readln;
end;

mmax:=0; {текущее значение мощности максимальной компоненты}
{строим связную компоненту от каждого элемента матрицы}
for k:=1 to N do
  for l:=1 to N do
    begin
      {инициализируем вспомогательный массив}
      for i:=0 to 11 do
        for j:=0 to 11 do
          b[i,j]:=0;
      b[k,l]:=1; {добавляем первый элемент компоненты}
      m:=1; {текущая мощность компоненты}
      mc:=1; {текущая мощность инкремента на шаге алгоритма}
      while mc>0 do {пока на предыдущем шаге были добавлены новые элементы}
        begin
          mc:=0;
          for i:=1 to N do
            for j:=1 to N do
              if (b[i,j]=0) then {элемент не добавлен в связную компоненту}
                if ((b[i,j-1]=1) and (a[i,j-1]=a[i,j]))
                  or ((b[i,j+1]=1) and (a[i,j+1]=a[i,j]))
                  or ((b[i-1,j]=1) and (a[i-1,j]=a[i,j]))
                  or ((b[i+1,j]=1) and (a[i+1,j]=a[i,j]))
                then {есть сосед в связной компоненте и значения ячеек совпадают}
                  begin {добавляем элемент в связную компоненту}
                    b[i,j]:=1;
                    inc(m);

```

```
        inc(m);  
    end;  
end;
```

```
if m>mmax then mmax:=m; {проверяем и переписываем максимум}  
end;
```

```
writeln(mmax); {Выводим ответ}  
end.
```