

**Министерство образования и науки Российской Федерации**

**Федеральное агентство по образованию**

**Государственное образовательное учреждение высшего профессионального образования**

**«Московский государственный технический университет имени Н.Э. Баумана»**

**(МГТУ им. Н.Э. Баумана)**

**Факультет «Робототехника и комплексная автоматизация» (РК)**

**Кафедра «Системы автоматизированного проектирования» (РК6)**

****

**Отчет по лабораторной работе №2 по курсу**

**«Методы комбинаторных вычислений»**

**Студент: Сергеева Диана**

**Группа:** РК6-56Б

**Преподаватель:** Волосатова Т.М.

Проверил:

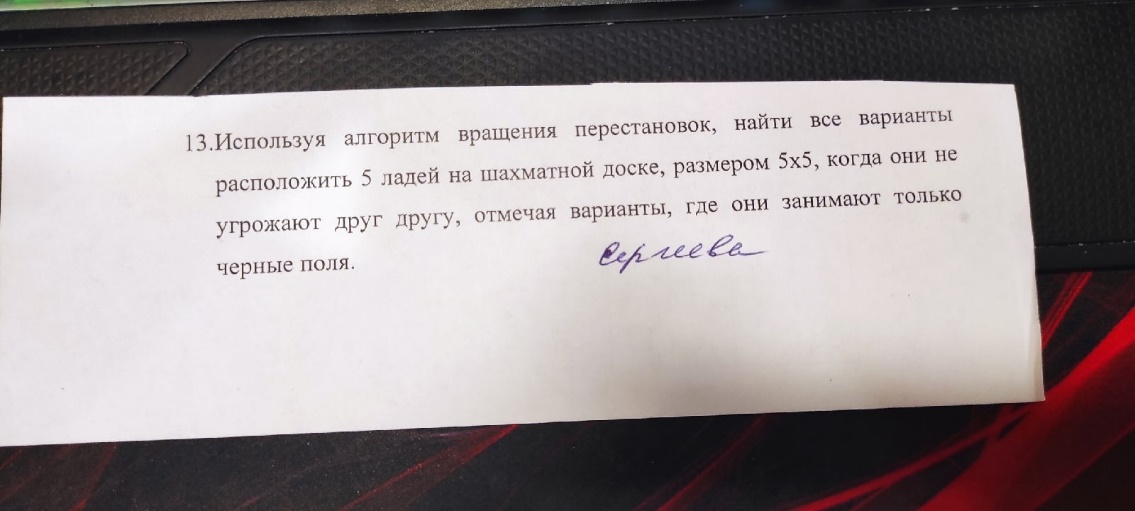
Дата:

2021 год

**13 вариант**

**Задание:**

Используя алгоритм вращения перестановок, найти все варианты расположить 5 ладей на шахматной доске, размером 5х5, когда они не угрожают друг другу, отмечая варианты, где они занимают только черные поля.



**Описание алгоритма:**

Алгоритм вращения основан на организации циклического сдвига элементов перестановки. Естественный способ перечисления перестановок циклическим сдвигом состоит в том, что, начав с некоторой исходной перестановки, нужно последовательно сдвигать по циклу на 1 позицию влево все n элементов. При каждом сдвиге 1-й элемент текущей перестановки перемещается на последнее место без изменения взаимного расположения остальных элементов, образуя новую перестановку. Такая организация цикла сдвига называется вращением.

Вращение всех элементов следует продолжать пока оно порождает оригинальные перестановки, не встречавшиеся ранее. Перестановка считается оригинальной, когда после сдвига позиция последнего элемента вращаемой части не равна его позиции в исходной перестановке.

Когда в результате очередного вращения образуется ранее порожденная перестановка, нужно исследовать возможность построить оригинальную перестановку, применяя вращение последовательно для первых j=n-1, n-2, ... ,2 элементов, при фиксированном положении (n-j) хвостовых элементов. Если локальное вращение 1<j<n элементов порождает оригинальную перестановку, следует продолжить вращение циклическим сдвигом всех n элементов. В противном случае (j=1), перебор считается завершенным, т. к. перечислены все n! перестановок.

Рассмотренный процесс систематического перечисления перестановок циклическим сдвигом называется алгоритмом вращения.

**Исходный код:**

#include <iostream>

#define N 5

using namespace std;

void printPermutation(int permutation[])

{

for (int i = 0; i < N; i++)

{

if (i == 0)

cout << "A";

else if (i == 1)

cout << "B";

else if (i == 2)

cout << "C";

else if (i == 3)

cout << "D";

else if (i == 4)

cout << "E";

cout << permutation[i] << " ";

}

}

void rotationPermutation(int permutation[], int n)

{

int firstNumPermutation = permutation[0];

for (int i = 0; i < n; i++)

{

permutation[i] = permutation[i + 1];

}

permutation[n] = firstNumPermutation;

}

int checkBlackCells(int permutation[])

{

int count = 0;

for (int i = 0; i < N; i++)

{

if ((((i + 1) % 2 == 0) && (permutation[i] % 2 == 0)) || (((i + 1) % 2 != 0) && (permutation[i] % 2 != 0)))

++count;

}

if (count == N)

return true;

else

return false;

}

void algorithmRotatingPermutations(int permutation[])

{

for (int i = 0; i < N - 3; i++)

{

for (int j = 0; j < N - 2; j++)

{

for (int k = 0; k < N - 1; k++)

{

for (int l = 0; l < N; l++)

{

printPermutation(permutation);

if (checkBlackCells(permutation))

cout << " - all black cells" << endl;

else

cout << endl;

rotationPermutation(permutation, N - 1);

}

rotationPermutation(permutation, N - 2);

}

rotationPermutation(permutation, N - 3);

}

rotationPermutation(permutation, N - 4);

}

}

int main()

{

int permutation[N];

for (int i = 0; i < N; i++)

{

permutation[i] = i + 1;

}

algorithmRotatingPermutations(permutation);

return 0;

}

**Результат работы программы:**

A1 B2 C3 D4 E5 - all black cells

A2 B3 C4 D5 E1

A3 B4 C5 D1 E2

A4 B5 C1 D2 E3

A5 B1 C2 D3 E4

A2 B3 C4 D1 E5

A3 B4 C1 D5 E2

A4 B1 C5 D2 E3

A1 B5 C2 D3 E4

A5 B2 C3 D4 E1 - all black cells

A3 B4 C1 D2 E5 - all black cells

A4 B1 C2 D5 E3

A1 B2 C5 D3 E4

A2 B5 C3 D4 E1

A5 B3 C4 D1 E2

A4 B1 C2 D3 E5

A1 B2 C3 D5 E4

A2 B3 C5 D4 E1

A3 B5 C4 D1 E2

A5 B4 C1 D2 E3 - all black cells

A2 B3 C1 D4 E5

A3 B1 C4 D5 E2

A1 B4 C5 D2 E3 - all black cells

A4 B5 C2 D3 E1

A5 B2 C3 D1 E4

A3 B1 C4 D2 E5

A1 B4 C2 D5 E3

A4 B2 C5 D3 E1

A2 B5 C3 D1 E4

A5 B3 C1 D4 E2

A1 B4 C2 D3 E5

A4 B2 C3 D5 E1

A2 B3 C5 D1 E4

A3 B5 C1 D4 E2

A5 B1 C4 D2 E3

A4 B2 C3 D1 E5

A2 B3 C1 D5 E4

A3 B1 C5 D4 E2

A1 B5 C4 D2 E3

A5 B4 C2 D3 E1

A3 B1 C2 D4 E5

A1 B2 C4 D5 E3

A2 B4 C5 D3 E1

A4 B5 C3 D1 E2

A5 B3 C1 D2 E4

A1 B2 C4 D3 E5

A2 B4 C3 D5 E1

A4 B3 C5 D1 E2

A3 B5 C1 D2 E4

A5 B1 C2 D4 E3

A2 B4 C3 D1 E5

A4 B3 C1 D5 E2

A3 B1 C5 D2 E4

A1 B5 C2 D4 E3

A5 B2 C4 D3 E1

A4 B3 C1 D2 E5

A3 B1 C2 D5 E4

A1 B2 C5 D4 E3 - all black cells

A2 B5 C4 D3 E1

A5 B4 C3 D1 E2

A2 B1 C3 D4 E5

A1 B3 C4 D5 E2

A3 B4 C5 D2 E1 - all black cells

A4 B5 C2 D1 E3

A5 B2 C1 D3 E4

A1 B3 C4 D2 E5

A3 B4 C2 D5 E1

A4 B2 C5 D1 E3

A2 B5 C1 D3 E4

A5 B1 C3 D4 E2

A3 B4 C2 D1 E5

A4 B2 C1 D5 E3

A2 B1 C5 D3 E4

A1 B5 C3 D4 E2

A5 B3 C4 D2 E1

A4 B2 C1 D3 E5

A2 B1 C3 D5 E4

A1 B3 C5 D4 E2

A3 B5 C4 D2 E1

A5 B4 C2 D1 E3

A1 B3 C2 D4 E5

A3 B2 C4 D5 E1

A2 B4 C5 D1 E3

A4 B5 C1 D3 E2

A5 B1 C3 D2 E4

A3 B2 C4 D1 E5

A2 B4 C1 D5 E3

A4 B1 C5 D3 E2

A1 B5 C3 D2 E4

A5 B3 C2 D4 E1

A2 B4 C1 D3 E5

A4 B1 C3 D5 E2

A1 B3 C5 D2 E4

A3 B5 C2 D4 E1

A5 B2 C4 D1 E3

A4 B1 C3 D2 E5

A1 B3 C2 D5 E4

A3 B2 C5 D4 E1 - all black cells

A2 B5 C4 D1 E3

A5 B4 C1 D3 E2

A3 B2 C1 D4 E5 - all black cells

A2 B1 C4 D5 E3

A1 B4 C5 D3 E2

A4 B5 C3 D2 E1

A5 B3 C2 D1 E4

A2 B1 C4 D3 E5

A1 B4 C3 D5 E2

A4 B3 C5 D2 E1

A3 B5 C2 D1 E4

A5 B2 C1 D4 E3 - all black cells

A1 B4 C3 D2 E5 - all black cells

A4 B3 C2 D5 E1

A3 B2 C5 D1 E4

A2 B5 C1 D4 E3

A5 B1 C4 D3 E2

A4 B3 C2 D1 E5

A3 B2 C1 D5 E4

A2 B1 C5 D4 E3

A1 B5 C4 D3 E2

A5 B4 C3 D2 E1 - all black cells

**Используемая литература:**

## Т.М. Волосатова, С.В. Родионов. Методы комбинаторных вычислений, учебное пособие. Издательство МГТУ им. Н.Э. Баумана. Москва. 2011