

## Чеканов К.Ю 774

### Формулировка решенных задач.

1) Решим задачу поиска кол-ва всевозможных полей в игре крестики-нолики, где под полем будем понимать состояние квадрата  $3 \times 3$  в какой-то игре в какой-то ее момент (ориентация игрового поля в пространстве учитывается).

2) Решим задачу нахождения всех финальных состояний поля (состояния поля  $3 \times 3$  на конец некоторой игры) не учитывая ориентацию поля в пространстве. (Промежуточная задача необходима для проверки корректности есть ответ на википедии <https://en.wikipedia.org/wiki/Tic-tac-toe>)

3) Решим задачу поиска кол-ва всевозможных полей в игре крестики-нолики не учитывая ориентацию поля в пространстве.

### Пункт 1. Комбинаторное решение.

На каждом уровне дерева игры вычислим кол-во всевозможных состояний поля.

**1 уровень:** Ставим крестик в одну из клеток **9**.

**2 уровень:** Ставим нолик в оставшиеся 8 клеток  $9 \cdot 8 = \mathbf{72}$ .

**3 уровень:** Ставим на поле два крестика их порядок не важен, и еще нолик  $C_9^2 \cdot 7 = \mathbf{252}$ .

**4 уровень:** Аналогично ставим 2 крестика, и два нолика их порядок не важен  $C_9^2 \cdot C_7^2 = \mathbf{756}$ .

**5 уровень:** Ставим на поле три крестика их порядок не важен, и еще 2 нолика, начнут появляться финальные состояние, но пока их не нужно убирать  $C_9^3 \cdot C_6^2 = \mathbf{1260}$ .

**6 уровень:** Аналогично всего расстановок  $C_9^3 \cdot C_6^3$ , из них  $8 \cdot C_6^3$  там где крестики уже встали в ряд, вычитаем их, получаем **1520**.

**7 уровень:** Всего расстановок  $C_9^4 \cdot C_5^3$ , из них  $8 \cdot C_6^3$  там где нолики уже встали в ряд  $C_6^4$ , вычитаем их, получаем **1140**.

**8 уровень:** Всего расстановок  $C_9^4 \cdot C_5^4$ , из них  $2 \cdot 8 \cdot C_6^4$  (можно доставить еще один крестик не входящий в тройку) там где крестики уже встали в ряд  $C_6^4$ , вычитаем их, получаем **390**.

**9 уровень:** Аналогично получаем **78**.

Суммируем полученные значения получаем: **5477**.

### **Пункт 1, 2, 3 программное решение.(Приложение: файл calculation.cpp)**

#### **Краткое описание алгоритма.**

Собственно, построим алгоритм, строящий дерево игры крестики нолики, если оценить его сверху оно имеет размер порядка  $2 \cdot 9! \approx 7 \cdot 10^5$ , что легко поддается вычислению. Реализуем этот алгоритм на C++, для уменьшения размера дерева будем хранить поле внутри переменной типа *int* (кодируем каждую ячейку поля двумя байтами), реализуем соответствующие функции и методы кодировки и декодировки, а также метод проверки поля на завершения игры. Классическим способом реализуем дерево в виде структуры значение-указатели. Далее очевидным полным перебором построим это дерево  $7C_9^2 + 2C_9^2 + 9$ . Имея это дерево далее анализируем его добавляя в контейнер *set* все поля с учетом и без учета ориентации.

#### **Ответ.**

Полученный ответ совпадает с результатом вышеприведенных вычислений, а также результат, полученный для пункта 2 совпадает со значением из википедии. На основании этого можно делать вывод о корректности работы программы.

```
kichyr@kichyr-HP-Notebook:~/Рабочий стол/tic-tac-toe$ ./a.out
All terminal fields, WITH the orientation(matches with wikipedia): 138
all fields WITHOUT the orientation: 5477
all fields WITHOUT the orientation: 764
```

#### **Подробнее про пункты 2, 3.**

Очевидно, что если не фиксировать положение доски в пространстве, то кол-во всевозможных полей на самом деле меньше. Учтем это, добавив всевозможные повороты и отражения, и получим ответ на пункты 2 и 3.