

Отчет

по лабораторной работе «1401. Игроки»
по дисциплине «Алгоритмы и структуры данных»

Авторы:

Полит Алексей Денисович

Факультет: СУиР

Группа: R3235

Преподаватель: Тропченко Андрей Александрович



УНИВЕРСИТЕТ ИТМО

1. Задача

Известно, что господин Чичиков зарабатывал свой капитал и таким способом: он спорил со всякими негодёнами, что сможет доказать, что квадратную доску размера 512×512 нельзя замостить следующими фигурами:

x	xx	x
xx		
xx	x	xx
x		

и всегда выигрывал. Однако один из негодён оказался не так уж глуп, и сказал, что сможет замостить такими фигурами доску размера 512×512 без правой верхней клетки. Чичиков, не подумав, ляпнул, что он вообще может любую доску размера $2n \times 2n$ без одной произвольной клетки замостить такими фигурами. Слово за слово, они поспорили. Чичиков чувствует, что сам он не докажет свою правоту. Помогите же ему!

2. Исходные данные

В первой строке записано целое число n ($1 \leq n \leq 9$). Во второй строке через пробел даны два целых числа x, y — координаты «выколотой» клетки доски ($1 \leq x, y \leq 2n$), x — номер строки, y — номер столбца. Левый верхний угол доски имеет координаты (1, 1).

3. Код программы

```
#include <iostream>
#include <cmath>

using namespace std;
int table[512][512];

void paint(int x, int y, int i, int j, int n) {
    static int c;
    ++c;
    n = n / 2;
    if ((x - j) >= n && (y - i) >= n) {
        table[i + n - 1][j + n - 1] = c;
        table[i + n][j + n - 1] = c;
        table[i + n - 1][j + n] = c;
        if (n > 1) {
            paint(j + n - 1, i + n - 1, i, j, n);
            paint(j + n, i + n - 1, i, j + n, n);
            paint(x, y, i + n, j + n, n);
            paint(j + n - 1, i + n, i + n, j, n);
        }
    } else if ((x - j) >= n && (y - i) < n) {
        table[i + n - 1][j + n - 1] = c;
        table[i + n][j + n - 1] = c;
    }
```

```

        table[i + n][j + n] = c;
        if (n > 1) {
            paint(j + n - 1, i + n - 1, i, j, n);
            paint(x, y, i, j + n, n);
            paint(j + n, i + n, i + n, j + n, n);
            paint(j + n - 1, i + n, i + n, j, n);
        }
    } else if ((x - j) < n && (y - i) < n) {
        table[i + n - 1][j + n] = c;
        table[i + n][j + n - 1] = c;
        table[i + n][j + n] = c;
        if (n > 1) {
            paint(x, y, i, j, n);
            paint(j + n, i + n - 1, i, j + n, n);
            paint(j + n, i + n, i + n, j + n, n);
            paint(j + n - 1, i + n, i + n, j, n);
        }
    } else if ((x - j) < n && (y - i) >= n) {
        table[i + n - 1][j + n - 1] = c;
        table[i + n][j + n] = c;
        table[i + n - 1][j + n] = c;
        if (n > 1) {
            paint(j + n - 1, i + n - 1, i, j, n);
            paint(j + n, i + n - 1, i, j + n, n);
            paint(j + n, i + n, i + n, j + n, n);
            paint(x, y, i + n, j, n);
        }
    }
}
return;
}

```

```

int main() {
    int n, x, y;
    cin >> n >> y >> x;
    n = 1 << n;
    if ((int) (pow(n, 2) - 1) % 3 != 0) {
        cout << -1;
        return 0;
    }
}

```

```

--x;
--y;
paint(x, y, 0, 0, n);

```

```

for (int i = 0; i < n; ++i) {
    for (int j = 0; j < n; ++j)
        cout << table[i][j] << " ";
    cout << endl;
}

```

```
}  
return 0;  
}
```

9328520	11:42:43 19 апр 2021	Aleksey	1401. Игроки	G++ 9.2 x64	Accepted		0.015	1 436 КБ
---------	-------------------------	-------------------------	------------------------------	-------------	----------	--	-------	----------

4. Описание алгоритма

Происходит проверка условия, необходимое для нужного размещения. Поле представлено в виде двумерного массива, куда записываются числа следующим образом: поле делится на 4 новых поля, которые меньше по размеру. На стыке трех полей, где не должна быть "дырка", размещается фигура. Процесс продолжается до тех пор, пока размер поля не станет 2*2. Вывод результатов.