| 🇬🇧 🇷🇺

TheKing003KS | Logout

You have +416! Wow!

HOME TOP CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP DELTIX ROUNDS 2021

Maintenance works on Round 739. Don't be surprised. Please, read the post. ×

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS ROOM STANDINGS CUSTOM INVOCATION

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
123025195	Virtual: TheKing003KS #	1530B - 10	GNU C++14	Accepted	15 ms	3828 KB	2021-07-19 17:17:02	2021-07-19 17:30:45	☆	Compare

→ Source

Copy

```
#include<iostream>
#include<limits.h>
#include<math.h>
#include<vector>
#include<string>
#include<queue>
#include<stack>
#include<set>
#include<map>
#include<unordered_set>
#include<unordered_map>
#include<algorithm>
using namespace std;

#define ll long long
#define ull unsigned long long
#define modulo 1000000007
#define mp make_pair
#define pb push_back

bool is_valid(int n, int m, int x, int y)
{
    return (x >= 0 && x < n && y >= 0 && y < m);
```

```
}

int main()
{
    ll int tests;
    cin >> tests;
    while(tests--)
    {
        int h, w;
        cin >> h >> w;

        if(w == 1)
        {
            bool flag = true;
            while(h > 0)
            {
                if(flag) {cout << 1 << "\n"; flag = false;}
                else {cout << 0 << "\n"; flag = true;}
                h--;
            }
        }
        else if(w == 2)
        {
            bool flag = true;
            while(h > 0)
            {
                if(flag) {cout << 1 << 0 << "\n"; flag = false;}
                else {cout << 0 << 0 << "\n"; flag = true;}
                h--;
            }
        }
        else
        {
            vector<vector<int>> ans(h,vector<int>(w,0));
            for(int j = 0; j < w; j += 2)
            {
                ans[0][j] = 1;
                ans[h-1][j] = 1;
            }
            if(h >= 5)
            {
                int i = 2;
                while(i <= h-3)
                {
                    ans[i][0] = 1;
                    ans[i][w-1] = 1;
                    i += 2;
                }
            }
            for(int i = 0; i < h; i++)
            {
                for(int j = 0; j < w; j++)
                {
                    cout << ans[i][j];
                }
                cout << "\n";
            }
        }

        cout << "\n";
    }

    return 0;
}

// ll int n;
// cin >> n;
// vector<ll int> arr(n);
// for(auto i = 0; i < n; i++) {cin >> arr[i];}

// for(auto i = 0; i < n; i++) {cout << arr[i] << " ";}

// sort(begin(arr),end(arr));
```

1

Time: 15 ms, **memory**: 3640 KB

Verdict: OK

Input

```
3
3 5
4 4
5 6
```

Participant's output

```
10101
00000
10101

1010
0000
0000
1010

101010
000000
100001
000000
101010

```

Jury's answer

```
10101
00000
```

```
10101

0100
0001
1000
0010

010101
000000
100001
000000
101010
```

Checker comment

```
ok ok, 3 test cases (3 test cases)
```

2

Time: 15 ms, **memory**: 3812 KB

Verdict: OK

Input

```
100
19 5
11 4
11 19
5 11
13 17
11 17
12 5
14 18
17 8
14 3
11 15
5 5
19 19
8 6
19 14
3 6
3 11
6 4
15 20
3 15
11 20
17 15
4 5
6 10
19 12
14 9
12 14
8 6
13 13
```

```
3 16
8 9
13 7
11 10
20 14
6 10
17 18
3 15
6 12
17 19
11 18
5 4
20 10
4 6
12 8
6 10
4 16
16 18
12 20
17 18
10 9
8 20
7 19
14 7
5 15
17 20
6 8
6 16
16 6
14 4
16 3
18 11
11 20
4 20
20 16
12 11
8 7
8 12
13 12
6 5
10 16
3 5
20 13
18 20
4 13
8 7
3 19
3 4
17 6
7 20
17 3
```

```
7 7
15 13...
```

Participant's output

```
10101
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10101

1010
0000
1001
0000
1001
0000
1001
0000
1001
0000
1010

1010101010101010101
0000000000000000000
1000000000000000001
0000000000000000000
1000000000000000001
0000000000000000000
1000000000000000001
0000000000000000000
1000000000000000001
0000000000000000000
1010101010101010101

10101010101
00000000000
10000000001
00000000000
10101010101
```

```
10101010...
```

Jury's answer

```
10101
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10001
00000
10101

1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001

1010101010101010101
0000000000000000000
1000000000000000001
0000000000000000000
1000000000000000001
0000000000000000000
1000000000000000001
0000000000000000000
1000000000000000001
0000000000000000000
1010101010101010101

10101010101
00000000000
10000000001
00000000000
10101010101
```

```
10101010...
```

Checker comment

```
ok ok, 100 test cases (100 test cases)
```

3

Time: 0 ms, **memory**: 3800 KB

Verdict: OK

Input

```
100
20 15
3 8
4 17
17 16
12 7
7 14
6 20
11 20
7 5
6 18
19 8
4 6
15 15
6 8
10 14
19 20
15 18
5 4
4 7
13 19
5 6
17 20
19 12
3 11
12 14
5 16
15 12
18 20
6 11
12 5
13 3
3 12
18 10
15 20
8 10
7 13
16 16
16 15
11 6
```

```
11 18
14 19
12 15
8 6
11 13
14 20
5 20
16 17
4 18
8 7
5 11
20 10
6 14
14 6
12 12
16 20
12 20
18 12
4 5
20 11
15 17
12 16
3 6
18 11
3 18
14 7
12 10
19 11
5 7
8 13
5 5
5 15
8 9
12 19
3 13
13 18
13 17
4 15
10 15
11 11
6 19
7...
```

Participant's output

```
101010101010101
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
```

```
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
000000000000000
101010101010101

10101010
00000000
10101010

10101010101010101
00000000000000000
00000000000000000
10101010101010101

1010101010101010
0000000000000000
1000000000000001
00000...
```

Jury's answer

```
101010101010101
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
100000000000001
000000000000000
000000000000000
101010101010101

10101001
00000000
10101001

10101010101010101
```

```
00000000000000000
00000000000000000
10101010101010101

1010101010101001
0000000000000000
1000000000000001
00000...
```

Checker comment

```
ok ok, 100 test cases (100 test cases)
```

4

Time: 0 ms, **memory**: 3808 KB

Verdict: OK

Input

```
100
15 3
14 16
6 6
7 7
11 9
15 14
6 13
14 11
15 11
19 7
9 7
10 6
14 15
5 3
7 6
6 12
18 14
10 7
12 3
13 16
17 6
11 17
20 8
10 10
13 14
18 13
4 8
7 12
3 19
18 15
10 19
4 19
11 3
```

```
3 10
8 18
13 12
8 11
12 9
3 4
5 10
5 17
8 17
18 3
19 15
13 8
16 5
13 10
3 16
19 10
17 8
3 9
13 9
7 3
19 17
8 19
20 7
8 4
10 4
9 3
17 10
11 5
20 13
3 3
20 9
10 11
17 3
6 10
19 9
7 17
12 8
11 15
8 15
4 12
18 16
11 12
16 3
6 16
14 18
5 14
13 6
3 15
20 20
...
```

Participant's output

```
101
000
101
000
101
000
101
000
101
000
101
000
101
000
101

1010101010101010
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
0000000000000000
1010101010101010

101010
000000
100001
000000
000000
101010

1010101
0000000
1000001
0000000
1000001
0000000
1010101

101010101
000000000
100000001
000000000
100000001
000000000...
```

Jury's answer

```
101
000
101
000
101
000
101
000
101
000
101
000
101
000
101

1010101010101001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
0000000000000000
1010101010101001

101001
000000
100001
000000
000000
101001

1010101
0000000
1000001
0000000
1000001
0000000
1010101

101010101
000000000
100000001
000000000
100000001
000000000...
```

Checker comment

```
ok ok, 100 test cases (100 test cases)
```

5

Time: 15 ms, **memory**: 3828 KB

Verdict: OK

Input

```
100
20 16
17 4
11 19
19 14
7 16
13 4
18 7
20 17
10 8
6 9
10 9
4 4
17 5
19 3
9 9
7 11
17 7
19 19
9 18
9 5
16 9
20 6
9 13
15 16
7 15
17 13
16 13
15 10
17 18
16 6
14 5
4 10
3 14
6 3
9 16
16 10
20 12
16 18
6 5
6 4
12 18
```

```
18 4
10 3
6 7
9 15
9 19
20 3
9 6
9 17
18 19
16 19
12 13
20 19
19 13
14 13
7 10
8 16
11 7
17 12
8 12
15 8
7 20
8 20
10 20
8 5
4 14
6 15
14 3
15 13
9 8
5 8
19 6
20 18
15 4
19 16
4 16
9 12
13 7
14 4
10 16
18 18
4 9
...
```

Participant's output

```
1010101010101010
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
```

```
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
0000000000000000
1010101010101010

1010
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1010

1010101010101010101
0000000000000000000
100...
```

Jury's answer

```
1010101010101001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
1000000000000001
0000000000000000
```

```
0000000000000000
1010101010101001

1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
1001

1010101010101010101
0000000000000000000
100...
```

Checker comment

```
ok ok, 100 test cases (100 test cases)
```

6

Time: 15 ms, **memory**: 3792 KB

Verdict: OK

Input

```
24
10 17
10 5
12 4
11 4
14 9
14 17
9 14
16 11
19 18
7 19
13 11
5 18
4 20
18 17
14 10
12 11
15 6
```

```
11 8
9 20
20 14
13 5
13 20
11 16
15 9
```

Participant's output

```
10101010101010101
00000000000000000
10000000000000001
00000000000000000
10000000000000001
00000000000000000
10000000000000001
00000000000000000
00000000000000000
10101010101010101

10101
00000
10001
00000
10001
00000
10001
00000
00000
10101

1010
0000
1001
0000
1001
0000
1001
0000
1001
0000
0000
1010

1010
0000
1001
0000
1001
0000
1001
0000
1001
```

```
0000
1010

101010101
000000000
100000001
000000000
100000001
000000000
100000001
000000000
100000001
000000...
```

Jury's answer

```
10101010101010101
00000000000000000
10000000000000001
00000000000000000
10000000000000001
00000000000000000
10000000000000001
00000000000000000
00000000000000000
10101010101010101

10101
00000
10001
00000
10001
00000
10001
00000
00000
10101

1001
0000
1001
0000
1001
0000
1001
0000
1001
0000
0000
1001

1001
0000
1001
```

```
0000
1001
0000
1001
0000
1001
0000
1001

101010101
000000000
100000001
000000000
100000001
000000000
100000001
000000000
100000001
000000...
```

Checker comment

```
ok ok, 24 test cases (24 test cases)
```

7

Time: 15 ms, **memory**: 3640 KB

Verdict: OK

Input

```
1
3 3
```

Participant's output

```
101
000
101
```

Jury's answer

```
101
000
101
```

Checker comment

```
ok ok, 1 test cases (1 test case)
```

8

Time: 0 ms, **memory**: 3648 KB

Verdict: OK

Input

```
4
5 5
5 6
6 5
6 6
```

Participant's output

```
10101
00000
10001
00000
10101

101010
000000
100001
000000
101010

10101
00000
10001
00000
00000
10101

101010
000000
100001
000000
000000
101010
```

Jury's answer

```
10101
00000
10001
00000
10101

010101
000000
100001
000000
101010

10101
00000
10001
00000
00000
10101
```

```
101001
000000
100001
000000
000000
101001
```

Checker comment

```
ok ok, 4 test cases (4 test cases)
```

9

Time: 0 ms, **memory**: 3780 KB

Verdict: OK

Input

```
100
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
```

```
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20 20
20...
```

Participant's output

```
10101010101010101010
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
```

```
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
00000000000000000000
10101010101010101010

10101010101010101010
00000000000000000000
10000000000000000001
000...
```

Jury's answer

```
10101010101010101001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
10000000000000000001
00000000000000000000
00000000000000000000
10101010101010101001

10101010101010101001
00000000000000000000
10000000000000000001
000...
```

Checker comment

```
ok ok, 100 test cases (100 test cases)
```

Supported by

ITMO UNIVERSITY