

Cloning socks

locked

Problem	Submissions	Leaderboard	Discussions
---------	-------------	-------------	-------------

Наскоро сте се сдобили с машина за клониране на чорапи. Знаете, че ако сложите нормален чифт чорапи в машината, тя ще ви даде още 1 нормален чифт и 1 копие на чорапите, но ако сложите копие на чорапите, тя ще ви даде още 2 копия.

Вие имате само 1 нормален чифт чорапи и спешно се нуждаете от още. По-точно, трябва ви точно N копия и M нормални чифта чорапи.

Дали е възможно да получите необходимия брой чорапи, използвайки машината за клониране?

Input Format

На първият ред на входа е дадено числото T. Следват T реда с по две числа на ред - N (необходим брой копия) и M (необходим брой оригинални чорапи).

Constraints

1 <= T <= 10

0 <= N, M <= 1 000 000 000

Output Format

Изведете T на брой реда. На всеки ред изедете "yes", ако е възможно да получите необходимия брой чорапи, или "no" в противен случай.

Sample Input 0

```
2
3 1
5 2
```

Sample Output 0

```
no
yes
```

Current Buffer (saved locally, editable)

C++

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;
void isCombinationPossible(vector < pair<int, int>> combinations) {
    int numberOfCombinations = combinations.size();
    for (int i = 0; i < numberOfCombinations; i++) {
        if(combinations[i].second==0){
            cout<<"no"<<endl;
        }
        else if (combinations[i].second >= combinations[i].first &&(combinations[i].second -
combinations[i].first>1)) {
            cout << "no" << endl;
        }
        else if (combinations[i].second == 1 && combinations[i].first != 0){
            cout << "no" << endl;
        }
        else if (combinations[i].second % 2 == 0 && combinations[i].first % 2 == 0) {
            cout << "no" << endl;
        }
        else if (combinations[i].second % 2 != 0 && combinations[i].first %2 != 0) {
            cout << "no" << endl;
        }
        else cout << "yes" << endl;
    }
}
int main() {
    int numberOfCombinations;
    cin >> numberOfCombinations;
    int normalSocks;
    int copySocks;
    vector<pair<int,int>> combinations;
    for (int i = 0; i < numberOfCombinations; i++) {
        cin >> normalSocks;
        cin >> copySocks;
        pair<int, int> Socks(normalSocks, copySocks);
        combinations.push_back(Socks);
    }

    isCombinationPossible(combinations);
    return 0;
}
```

Line: 1 Col: 1