780. Reaching Points

DescriptionHintsSubmissionsDiscussSolution

- Difficulty:Hard
- Total Accepted:1.3K
- Total Submissions:6.8K
- Contributor:awice
- Related topic : Recursion
- Subscribe to see which companies asked this question.

A move consists of taking a point `(x, y)` and transforming it to either `(x, x+y)` or `(x+y, y)`.

Given a starting point `(sx, sy)` and a target point `(tx, ty)`, return `True` if and only if a sequence of moves exists to transform the point `(sx, sy)` to `(tx, ty)`. Otherwise, return `False`.

```
Examples:
Input: sx = 1, sy = 1, tx = 3, ty = 5
Output: True
Explanation:
One series of moves that transforms the starting point to the target is:
(1, 1) -> (1, 2)
(1, 2) -> (3, 2)
(3, 2) -> (3, 5)

Input: sx = 1, sy = 1, tx = 2, ty = 2
Output: False

Input: sx = 1, sy = 1, tx = 1, ty = 1
Output: True
```

Note:

- `sx, sy, tx, ty` will all be integers in the range `[1, 10^9]`.

```
class Solution {

public:

bool reachingPoints(int sx, int sy, int tx, int ty) {

    if(sx==tx && sy==ty) return true;

    else if (tx==ty || sx>tx || sy>ty) return false;

    else if (tx>ty)
```

```
    {
          int multiple = max(1,(tx-sx)/ty);
          return reachingPoints(sx,sy,tx - multiple*ty,ty);
     }
     else if (ty>tx){
          int multiple = max(1,(ty-sy)/tx);
          return reachingPoints(sx,sy,tx,ty-multiple*tx);
     }
}

};
```