

Summer Institute '17 Contest 1 P3 - CEO Queue

Time limit: 1.4s **Memory limit:** 256M
C: 0.75s
C++: 0.75s

Summer Institute @ University of Central Florida: Contest 1, Problem 3

The Grandestine made a two player online game similar to chess but with more pieces and free-to-play mechanics to rake in the bucks. Each player is given a rating that starts out at 0 and goes up or down as they win or lose games. Grand implemented a queue system that attempts to pair up players with similar ratings. Whenever a player enters the queue the game creates a range centered around that player's rating that expands over time which specifies acceptable opponent ratings. The range expands by 1 in each direction per unit time. For example, if a player rated 2050 entered the queue at time 0, at time 40 that player's range will be from 2010 to 2090. For a pair of players to be matched up, at least one of the players' ratings must fall within their opponent's rating range. Expanding on the previous example, if another player entered the queue at time 70 with rating 2150 he could be paired up with the first player at time 100 when the first player's range is 1900 to 2150 and the second player's range is 2120 to 2180. If instead the second player had a rating of 1990 he would have been immediately paired with the first player when he entered the queue at time 70 when the first player's range was 1980 to 2120. The queue pairs up players as soon as possible. In the event that multiple pairs of players can be matched at the same time, priority is given to the pair that contains the player who has been in the queue the longest. If there are multiple pairs that share this player, the tiebreaker is how long the other player of the pair has been in the queue. It is guaranteed that players do not join the queue at the same time and that no two players share the same rating.

Once a player is paired up for a match, the player will not be placed back into the queue. Thus, all players play either 0 or 1 match.

Grand is interested in the queue wait times for the players. In particular he wants to know how many players spent at least w time units waiting in the queue without being matched.

Input Specification

The first line of input contains two space separated integers n ($1 \leq n \leq 10^5$) and w ($1 \leq w \leq 10^8$), the number of players that will join the queue and the number of minutes for the input query. The following n lines contain information about each player. The i th of these lines contains two space separated integers: q_i ($0 \leq q_i \leq 10^8$) and r_i ($0 \leq r_i \leq 10^8$), the time the i th player enters the queue and the i th player's rating, respectively. The players are given in strictly increasing order of queue times.

Output Specification

Output a single integer, the number of players that waited at least w time units in the queue without being matched.

Sample Input 1

```
3 10
10 1000
11 1200
50 1001
```

Sample Output 1

```
2
```

Explanation for Sample 1

The players rated 1000 and 1001 get matched up at time 50. The player rated 1200 is stuck in the queue forever.

Sample Input 2

```
6 200
0 0
20 1000
30 510
40 400
50 300
320 500
```

Sample Output 2

```
4
```

Explanation for Sample 2

- The players rated 510 and 400 get paired at time 140
- The players rated 0 and 300 get paired at time 300
- The players rated 1000 and 500 get paired at time 520