Problem 3 - IOICamp (Programming) (15 points)

Problem Description

IOICamp is a competitive programming training camp held by NTU CSIE students. Xiao Feng, an IOICamp staff, is full of enthusiasm to dedicate himself to making it better.

One day, Baluteshih, the general coordinator of IOICamp, assigned N tasks to Xiao Feng. Concretely, the i^{th} task consists of x_i units of work. Each unit of work costs one day for Xiao Feng to complete. Besides, Xiao Feng can not do two different tasks in one day simultaneously. Because of Baluteshih's requirements, Xiao Feng can only work on the i^{th} task between ${s_i}^{\text{th}}$ day and ${e_i}^{\text{th}}$ day (inclusive).

To encourage Xiao Feng to finish as much work as possible, Baluteshih gives him some benefits. For the i^{th} task, if Xiao Feng completes y_i units of work, he will get $p_i \cdot y_i$ bonus from Baluteshih. Note that Xiao Feng can ~~still get the bonus~~ from Baluteshih even if he didn't finish all units of work in the one task. Please help Xiao Feng to calculate the maximum bonus he could receive from his kindly boss, Baluteshih :).

Input

The first line of the input contains only one integer N, denoting the number of tasks assigned to Xiao Feng.

In the following N lines, the i^{th} line contains four integers s_i, e_i, x_i, p_i, describing the information of the i^{th} task.

constraints

- $1 \le N \le 3000$.
- $1 \le s_i \le e_i \le 10^9$.
- $1 \le x_i \le e_i - s_i + 1$.
- $1 \le p_i \le 10^9$.

Test Group 0 (0 %)

- Sample Input.

Test Group 1 (15 %)

- $s_1 \le s_2 \le \ldots \le s_N$.
- $e_1 \le e_2 \le \ldots \le e_N$.
- $p_i = 1, \forall 1 \le i \le N$.

Test Group 2 (40 %)

- $p_i = 1, \forall 1 \le i \le N$.

Test Group 3 (15 %)

- $s_1 \le s_2 \le \ldots \le s_N$.
- $e_1 \le e_2 \le \ldots \le e_N$.
- $x_i = e_i - s_i + 1, \forall 1 \le i \le N$.

Test Group 4 (30 %)

- No other constraints.

Output

Print one integer denoting the maximum bonus Xiao Feng could receive if he arranges the work optimally.

Sample Input 1

s_i l_i x_i p_i

```
3
1 3 2 1
1 5 1 1
2 4 1 1
```

Sample Output 1

```
4
```

Sample Input 2

```
5
6 7 2 6
1 10 3 6
6 8 2 8
3 8 1 9
1 9 7 2
```

Sample Output 2

```
55
```

Sample Input 3

```
5
9 10 1 5
5 15 6 7
4 6 2 8
1 6 1 3
3 9 1 1
```

Sample Output 3

```
67
```

Sample Input 4

```
10
317828572 952962709 511194031 474210
139065667 594136128 184836056 727043
145449199 856665845 135232964 221941
185367317 719253355 508496356 303732
286924029 536237215 174723858 743784
448407424 788782769 294918233 970051
128701901 369779350 133590454 996886
268148730 724234276 442825804 255091
658359136 999211180 190588357 715619
114934339 328552693 120729904 373197
```

Sample Output 4

```
741483180481768
```

Hint

1. It might be useful to solve **Test Group 2** before solving the rest of the problem. If you encounter some difficulties, please try to solve it first.
2. It is recommended to use C++ Standard Template Library (STL) data structures, such as `std::stack`, `std::queue`, `std::priority_queue`, `std::list`, `std::vector`, and `std::set`. They can reduce your coding complexity.