

Salve as pizzas do IC

Trabalho 2 de MC458 - Entrega: 22/06

Finalmente chegou o dia de um dos eventos mais esperados pelos estudantes da universidade: a pizzada do Instituto de Culinária (IC).

Foram preparadas N pizzas para o evento. Todas as pizzas do IC são feitas com ingredientes frescos de alta qualidade, o que as proporciona um alto nível de sabor. Em contrapartida, à medida que o tempo passa o frescor dos ingredientes diminui, consequentemente comprometendo o sabor final da pizza, podendo até perder todo o sabor caso demore tempo demais para assá-la.

Com isso, sabe-se que cada pizza i leva um tempo t_i para ser assada, possui sabor inicial s_i , e perde r_i de sabor por minuto, até eventualmente chegar em 0. A pizza para de perder sabor apenas no fim do preparo. Isto é, se a pizza i começar a ser assada no tempo t , então o seu sabor final será $s_i - r_i(t + t_i)$, e não $s_i - r_i t$.

Está próximo do início do evento e todas as pizzas estão preparadas e os fornos já estão pré-aquecidos. Porém, logo antes de começar a assar as pizzas um desastre aconteceu: uma árvore caiu no IC, danificando o funcionamento de todos os fornos, exceto um. Agora só é possível assar uma pizza de cada vez, e restam apenas T minutos para o início do evento. Diante da situação, você deve rapidamente escolher dentre as N pizzas quais assar nos T minutos restantes e em que ordem, de forma que a soma dos sabores finais das pizzas escolhidas seja o maior possível.

Entrada

A primeira linha contém os inteiros N e T . Seguem N linhas, onde a i -ésima linha contém os valores de s_i , t_i e r_i , nessa ordem.

Saída

Uma linha contendo a maior soma de sabores finais possível.

Exemplo

Entrada	Saída
3 60 80 25 1 70 20 2 60 15 3	65

Neste exemplo, a solução ótima consiste em assar primeiro a pizza 2, que terá sabor final $70 - 20 \cdot 2 = 30$, seguida da pizza 1, que terá sabor final $80 - 45 \cdot 1 = 35$, totalizando 65.

Restrições

- $1 \leq N, T \leq 1000$;
- $1 \leq s_i \leq 1000$;
- $1 \leq r_i \leq 50$;
- 50% dos casos de teste: $t_i = 1$;
- 50% dos casos de teste: $1 \leq t_i \leq 50$;
- Tempo limite de execução: 1 segundo.

Sua nota será dada pela quantidade de casos de teste corretos e dentro do tempo limite que seu algoritmo é capaz de resolver. Note que os testes não vão apenas verificar corretude, mas também a complexidade de tempo para resolver o teste. Assim, programas corretos mas com complexidade de tempo alta provavelmente não resolverão a tempo os testes maiores. Para obter uma estimativa da complexidade de tempo esperada, pode-se estimar que a máquina é capaz de realizar até uma quantidade da ordem de 10^7 instruções dentro de 1s.

Instruções para submissão

- Você deve entregar um código `.c` ou `.cpp`, além de um relatório que explica seu algoritmo.
- A entrega do código será feita pelo SuSy, pelo link <https://susy.ic.unicamp.br:9999/mc458a/2>.

- O código pode ser escrito em C ou C++. Serão usados os parâmetros de compilação `-std=c99 -pedantic -Wall -lm` para C, e `-ansi -pedantic -Wall -lm` para C++.
- É permitido um total de 20 submissões. **Apenas a submissão mais recente será considerada.**
- A entrega do relatório será feito pelo Classroom, na tarefa correspondente, através de um documento em pdf.
- Mesmo que sua solução passe em todos os casos de teste no Susy, algoritmos que não tenham a complexidade de tempo esperada ainda estarão sujeitas a perder pontos.