

Pobieranie

Jaś chce pobrać dużą liczbę plików (n - liczba plików do pobrania). Jednak ma ograniczoną prędkość pobierania (m - prędkość pobierania w Mb/s), z tego powodu ustalił priorytet pobierania poszczególnych plików (p - priorytet pobierania pliku, najpierw pobierają się pliki które mają priorytet nr 1 gdy już wszystkie z tym priorytetem się pobiorą wtedy zaczynają się pobierać pliki z priorytetem nr 2, i tak dalej aż do końca). Jednak Jaś chciał żeby pliki o tym samym priorytecie nie pobierały się z tą samą prędkością więc dla każdego pliku ustalił jeszcze liczbę q (q - jest to udział danego pliku w wykorzystaniu łącza, im wyższa tym szybciej dany plik się pobiera np. jeśli plik A ma udział 1, a plik B ma udział 3, to plik A pobiera się z prędkością $0,25 * m$, a plik B z prędkością $0,75 * m$, gdy któryś pobierze się wcześniej wtedy udział w pobieraniu jest rozdzielany na pozostałe pliki). Jaś chce wiedzieć kiedy zakończy się pobieranie poszczególnych plików, lecz ten problem przerósł go, napisz dla Jasia program który to obliczy.

Input

W pierwszej linii są podane dwie liczby n, m oddzielone spacją. Następnie n linii opisujące poszczególne pliki. Każda linia składa się z liczb: nr p q r , (nr - numer porządkowy pliku, p - priorytet danego pliku, q - udział w prędkości pobierania danego pliku, r - rozmiar danego pliku w Mb). Wielkości liczb to:

$1 \leq n \leq 100$

$1 \leq p \leq nr \leq n$

$1 \leq m, r, q \leq 1000000$

Output

Wyjście składa się z n linii w które składają się z dwóch liczb: pierwsza to nr danego pliku, druga liczba to czas (w sekundach) po jakim zakończy się pobieranie danego pliku. Dane powinny być uporządkowane w zależności od czasu zakończenia pobierania danego pliku.

Example

Input:

```
5 10
1 1 1 10
2 1 3 40
3 2 5 50
4 3 1 20
5 3 1 10
```

Output:

```
1 4
2 5
3 10
5 12
4 13
```

// testy są tak dobrane że wszędzie podczas obliczania są liczby całkowite

