

## iPeds

P97568\_es

El nuevo *gadget* se llama iPed: los consumidores se lanzan desesperados a conseguirlo, y la fábrica de China que los produce no da abasto. Tan pronto tienen las cuatro piezas necesarias para montar un iPed (la Carcasa, la Pantalla, la Batería y el Microcontrolador), uno de sus empleados ensambla el iPed resultante sin perder ni un solo segundo de tiempo.

Sabiendo cuándo llegan las distintas remesas de componentes a la fábrica, escribe un programa que calcule el número de iPeds que podrán fabricarse, y cuándo estarán disponibles.

### Entrada

La primera línea de la entrada contiene el número  $n$  de remesas que recibe la fábrica, seguido de una cantidad arbitraria de líneas con las  $n$  remesas (pueden haber varias remesas en la misma línea). Cada remesa es un triplete de valores con el instante  $t \geq 0$  en el que llega la remesa, el número de componentes  $m > 0$  que contiene, y el tipo (C, P, B y M) de los mismos.

### Salida

Siempre que sea posible ensamblar un nuevo iPed, escribe una línea con el instante  $t$  y el número total de nuevos iPeds que puedan ensamblarse en ese instante. Escribe las líneas en orden cronológico, y no escribas dos líneas con el mismo valor instante  $t$ .

### Puntuación

- **Test1:** 30 Puntos  
Resolver entradas todas las remesas se dan en orden creciente en función del tiempo  $t < 1000$  de llegada, todos los instantes de llegada son diferentes, y todas las remesas contienen un único componente (o sea,  $m = 1$  siempre), como en el Ejemplo 1.
- **Test2:** 25 Puntos  
Resolver entradas donde  $t, m, n < 10^4$ , como el Ejemplo 2.
- **Test3:** 35 Puntos  
Resolver entradas donde  $t < 10^9$ ,  $m, n < 10^4$ , como el Ejemplo 3.
- **Test4:** 10 Puntos  
Resolver entradas donde  $t < 10^9$  y  $m, n < 10^5$ .

### Ejemplo de entrada 1

```
31
50 1 C 51 1 B
60 1 P 65 1 M
100 1 C 101 1 C
102 1 B 103 1 M
110 1 B 111 1 P
112 1 C 120 1 P
150 1 C 200 1 M
210 1 P 212 1 C
215 1 B 218 1 B
225 1 M 228 1 P
229 1 P 235 1 C
238 1 C 242 1 M
243 1 B 246 1 M
254 1 M 257 1 M
260 1 M 299 1 B
```

300 1 B

### Ejemplo de salida 1

65 1  
111 1  
200 1  
225 1  
242 1  
246 1

### Ejemplo de entrada 2

13  
50 10 C 50 5 B  
60 6 P 60 2 M  
200 40 C  
500 33 M 500 20 M  
400 71 P  
300 84 B  
600 100 C 600 5 M 600 10 P  
500 1 C

### Ejemplo de salida 2

60 2  
500 49  
600 9

### Ejemplo de entrada 3

30  
273951903 8001 P 786053619 6693 C  
473050900 788 M 89070091 3605 M  
663155708 6493 C 73292730 4871 M  
925768777 827 B 175399328 2633 B  
512713241 3125 P 425533345 1914 P  
223117608 2770 M 71022711 816 B  
273951903 5001 C 786053619 8696 P  
473050900 6322 B 89070091 6178 P  
663155708 9534 M 73292730 35 C  
925768777 4962 B 175399328 5781 P  
512713241 7723 P 425533345 7679 C  
223117608 4724 B 71022711 4218 B  
71022711 8163 C 71022711 81 M  
71022711 816 P 71022711 8 P  
71022711 84 P 71022711 100 P

### Ejemplo de salida 3

71022711 81  
73292730 927  
89070091 4026  
175399328 2633  
223117608 531  
273951903 3129  
473050900 788  
663155708 6598  
925768777 2936

### Información del problema

Autoría: Omer Giménez

Generación: 2026-01-25T12:14:57.594Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>