

---

## El Dilema dels Cristalls de Dilithi

Y10201\_ca

---

El Tinent Comandant Geordi La Forge, cap d'enginyeria de la USS Enterprise, es troba amb un repte important. Disposa de  $N$  cristalls de dilithi, el poderós combustible dels motors de curvatura. Pot utilitzar aquests cristalls per crear configuracions energètiques de diverses potències: amb un cristall, o dos, o tres... per configuració, fins a  $N$  per a la configuració més potent.

L'eficiència de cada configuració varia segons factors com l'estabilitat quàntica, la ressonància subatòmica i les interferències amb el camp de contenció. Geordi té dades precises sobre l'eficiència (mesurada en unitats de potència warp) que s'obté amb cada configuració, des d'1 fins a  $N$  cristalls. Curiosament, l'eficiència no sempre és proporcional: algunes configuracions amb pocs cristalls poden ser sorprenentment eficients, mentre que d'altres amb molts cristalls poden presentar problemes d'estabilitat que en redueixen el rendiment.

Abans que la Enterprise es trobi amb els Romulans, Geordi necessita distribuir els  $N$  cristalls disponibles entre diferents configuracions per maximitzar la potència total i estar preparat per a qualsevol eventualitat.

En Geordi haurà de fer una funció `reparteix_cristalls(lst)` on `lst` és una llista amb les eficiències ordenades, és a dir, `lst[i]` conté l'eficiència de fer servir configuracions amb  $i+1$  cristalls.

Exemple: Si Geordi disposa de 7 cristalls de dilithi, i les eficiències (en unitats de potència warp) són: 1: 10; 2: 12; 3: 16.1; 4: 35.8; 5: 69; 6: 75.33; 7: 88, la potència màxima que pot aconseguir és 89 unitats; ho aconsegueix creant dues configuracions amb 1 cristall i una configuració amb 5 cristalls. És a dir, el resultat de fer la crida `reparteix_cristalls([10, 12, 16.1, 35.8, 69, 75.33, 88])` és 89.

### Entrada

La funció té 1 paràmetre, `lst`, que és una llista no buida d'eficiències, on `lst[i]` conté l'eficiència de fer servir grups de  $i+1$  cristalls.

### Sortida

La potència màxima que pot obtenir Geordi per als motors de curvatura (*tot i que no s'hagi d'especificar com s'assoleix aquesta potència per a la correcció automàtica, el corrector humà valorarà positivament que el programa pugui mostrar aquesta informació fàcilment*).

### Observacions

Cal aplicar un esquema de programació dinàmica.

Un cop definida la funció, en provar-la al REPL de Python us hauria de sortir el mateix que podeu observar més avall.

### Informació del problema

Autoria: Jordi Delgado

Generació: 2026-01-25T19:54:49.638Z

© *Jutge.org*, 2006–2026.  
<https://jutge.org>