
La bossa

P50045_ca

Heu d'implementar una classe per a una "bossa d'enters" amb les operacions següents:

- `init`: Crear una bossa buida amb paràmetre k (amb $k \geq 0$).
- `add`: Afegir un enter a una bossa. Es poden afegir fins a k repeticions de cada enter: Quan la bossa contingui k repeticions d'un cert enter, afegir aquell enter no canvia la bossa.
- Esborrar (`remove_minimum`) i obtenir (`minimum`) l'element més petit de la bossa (si no és buida). L'operació d'esborrar només esborra una de les ocurrences de l'element mínim en cas de tenir-ne repeticions.
- Saber quants elements hi ha a la bossa comptant possibles repeticions (`size`) i quants elements diferents (sense repeticions) hi ha a la bossa (`items`).
- Saber si la bossa és buida (`empty`).

Descarregueu-vos el fitxer `code.py` i anomenau-lo `bag.py`. Aquest ja conté la interfície de la classe i un programa principal de proves que la fa servir.

Doneu el cost asimptòtic de cadascuna de les operacions públiques en funció d' n on n és el nombre d'elements a la bossa; podeu suposar que k és un valor constant. Comproveu els possibles errors amb assercions. La vostra implementació ha de ser eficient: cap operació pot tenir cost superior a logarítmic. Documenteu el vostre codi adequadament (ni poc, ni massa).

Exemple d'entrada 1

```
3
empty
size
items
add 22
add 22
add 22
add 22
empty
size
items
```

Exemple de sortida 1

```
True
0
0
False
3
1
```

Exemple d'entrada 2

```
3
add 1 add 1 add 1 add 1
add 2 add 2 add 2 add 2
add 3 add 3 add 3 add 3
add 4 add 4 add 4 add 4
size
items
minimum
remove_minimum
remove_minimum
remove_minimum
remove_minimum
minimum
size
items
add 1 add 1 add 1 add 1
minimum
size
items
empty
```

Exemple de sortida 2

```
12
4
1
1
1
1
1
2
2
8
3
1
11
4
False
```

Informació del problema

Autor : Jordi Petit

Generació : 2025-04-04 08:31:42

© *Jutge.org*, 2006–2025.

<https://jutge.org>