

---

## Eliminar repetits en una pila

X19470\_ca

---

Implementeu una funció **RECURSIVA** que, donada una pila  $P$  com a paràmetre (ordenada de fons cap amunt, i potser amb repeticions), retorna una pila  $R$  que conté tots els valors que hi ha a  $P$  sense repeticions. La funció és aquesta:

```
stack<int> uniq(stack<int> s);
```

PRE  $P = [a_1, a_2, \dots, a_n]$  i  $\forall_{i=1}^{n-1} a_i \leq a_{i+1}$ .

POST Torna una pila  $R = [b_1, b_2, \dots, b_m]$  tal que  $\forall_{i=1}^m b_i < b_{i+1}$  i a més  $\forall_{i=1}^m a_i \in R$ . És a dir,  $R$  conté tots els valors que hi ha a  $P$  sense repeticions i en el mateix ordre.

Aquí tenim uns exemples de comportament de la funció:

```
uniq([1 2 2 2 3 5 5 5 5]) = [1 2 3 5]
```

```
uniq([1 2 3 4]) = [1 2 3 4]
```

### Observació

Només cal enviar el procediment demanat; el programa principal serà ignorat.

### Observació

La vostra funció i subfuncions que creeu han de treballar només amb piles. Heu de trobar una solució **RECURSIVA** i eficient del problema. En particular, no hi hauria d'haver cap bucle en cap de les funcions que implementeu. Si creeu funcions auxiliars, afegiu-hi les seves PRE/POST. En les crides recursives, incloeu la funció de fita/decreixement.

Avaluació sobre 10 punts:

- Jocs de prova públics, semàfor verd: 6 punts.
- Jocs de prova privats, semàfor verd: 4 punts

Coses que poden restar punts a la puntuació anterior:

- Recursiu en comptes d'iteratiu (o viceversa): zero de nota final.
- Utilització d'estructures de dades auxiliars diferents del tipus que apareix a l'enunciat: des de -5 fins a zero de nota final. En cas de dubte, demaneu al professor.
- No fer immersió de paràmetres/resultats si això millora el cost asimptòtic del vostre codi: de -5 cap endavant, depenent de la gravetat.
- Manipulació *excessiva* d'estructures de dades: de -5 cap endavant. Exemple: agafar una pila i capgirar-la més del que cal.

- No posar PRE/POST a les funcions auxiliars que feu servir: -3. Òbviament, ja us donem la PRE/POST de la funció que us demanem, no cal que la repetiu. Ara bé, si feu una funció que implementa una generalització de la funció que us demanem, sí que cal que en feu la PRE/POST.
- No posar l'invariant si feu un bucle: -2.

### **Informació del problema**

Autor :

Generació : 2023-10-27 12:50:21

© *Jutge.org*, 2006–2023.

<https://jutge.org>