

Pràctica 1. Estructures de dades bàsiques. Piles i cues.

Pràctica 1

Estructura de Dades – Universitat de Barcelona
Rubén Ballester Bautista – Oriol Rabasseda Alcaide

NIUB – Nom:

16799506 – Rubén Ballester Bautista

16856733 – Oriol Rabasseda Alcaide

Grup de Pràctiques F – Parella **1**

Professora:

Maria Salamó

Exercicis comentats

Exercici 1:

Com has comprovat amb una pila que l'expressió està ben aparellada? Quins errors controles i com ho fas?

-Primer hem creat una pila únicament amb els elements aritmètics que controlen l'ordre de la expressió (parèntesis, claudàtors, claus...). Després, procedim a comprovar si la pila està buida. Si ho està doncs està bé i acaba, si no procedim a executar el següent algorisme:

1. Creem una nova pila buida anomenada «pilaAux2». La original la anomenarem «pila».
2. Si «pilaAux2» està buida, transferim el últim caràcter de «pila» a «pilaAux2» i l'eliminem de «pila». Anem directament al pas 5.
3. Comprovem si el top de «pilaAux2» és igual al element oposat del top de «pila» (Nota: únicament comprovem en cas que sigui una expressió de tancada d'expressió)- Si és igual esborrem els elements dos de les dues piles i anem directament al pas 5. Si no es cert anem al pas 4.
4. Transferim el element oposat de «pila» a «pilaAux2». Anem al pas 5
5. Si la pila no està plena tornem a reiterar el procés a partir del pas 2.

Podem demostrar que el loop acaba. Si la mida de la pila és n i a més a més la mida de la pila és un enter positiu, açò implica que tenim una mida n que segueix la següent regla: $0 \leq n$. Si $n=0$ hem acabat l'algorisme, si no, decrementem la mida en 1 per cada iteració, i per tant tenim una mida de $n-1$ on n és la mida actual. Si $n-1 = 0$ la pila és buida i el algorisme acaba, si no repetim el procés fins que $n' = 0$. QED

Per finalitzar, si pila 2 és buida \Rightarrow la expressió és correcta.

La complexitat de l'algorisme és d' $O(n)$. Açò es perquè tenim $O(n)$ de crear la primera pila + $O(n)$ de comprovar tots els elements de la primera pila + $O(1)$ (sentències constants) $\Rightarrow f(n) = 2 \times O(n) + O(1) \in O(n)$

Els errors que controlem són:

- Pila buida (mètodes pop(), top())
- Pila plena (mètodes push(char c))

Exercici 2:

Quina és la sortida del main?

Mida actual de la cua: 0

Encuem 3 elements a la cua...

1

2

3

Cua plena (0: no, 1: si): 1

Traiem 1er element de la cua: 1

2

3

Traiem 2on element de la cua: 2
Encuem 2 elements a la cua...
Traiem 3er element de la cua: 3
Mida actual de la cua: 2
4
5
Cua buida (0:no, 1:si): 0

Quins errors controleu dels que es poden produir quan es demana afegir o eliminar elements de la cua?

- Afegir elements a la cua: Excepció de cua plena
- Suprimir element de la cua: Excepció de cua buida.

Què feu per controlar els errors i com ho feu?

- Cua plena. Si sentinella de mida de cua = mida màxima => llancem excepció de «out_of_range» (cua plena).
- Cua buida. Si sentinella de mida de cua = 0=> llancem excepció de «out_of_range» (cua buida).

Exercici 3:

Hem implementat els mateixos algoritmes que a l'exercici 1 però amb templates de C++.