

SI-möte #9, Programkonstruktion och Datastrukturer

Elias Castegren & Kristiina Ausmees

elca7381@student.uu.se || krau6498@student.uu.se

23 januari 2012

Begrepp

- i)* Vad är en *stack*? Vilka är de två grundläggande stackoperationerna?
- ii)* Vad menas med en *FIFO-kö*? Hur skiljer sig en kö från en stack?
- iii)* Vad är komplexiteten för lägga till eller ta bort ett element från en stack respektive en kö om man representerar dem med enkelt länkade listor (vanliga listor i ML)?
- iv)* Vad menas med en *abstrakt datatyp* (*abstype*)? Vad menas med en abstrakt datatyps *gränssyta*?
- v)* Vad skiljer en abstrakt datatyp från en vanlig datatyp (*datatype*)? Vad finns det för fördelar med att använda en abstrakt datatyp?

Övningar

1.

Utför följande insättningar och borttagningar till en (tom) stack. Skriv ut hela stackens utseende i varje steg.

- i)* Pusha 9, 5, 1, 2 och 3
- ii)* Poppa två element
- iii)* Pusha 4, 1, 3, 7
- iv)* Poppa ett element

Utför samma operationer fast till en (FIFO-)kö, alltså med enqueue och dequeue istället för push och pop. Skriv ut hela köns utseende i varje steg.

2.

Skriv en abstrakt datatyp `"a set` (med konvention och invariant) som representerar mängder. Kom ihåg att en mängd inte innehåller några dubletter! Låt en mängd internt representeras som en lista. Gränsytan ska innehålla följande värden och funktioner:

- `empty` - Ett värde som är en tom mängd.
- `member(e, S)` - `true` om $e \in S$, annars `false`.
- `add(e, S)` - En ny mängd $\{e\} \cup S$
- `remove(e, S)` - En ny mängd $S - \{e\}$
- `cardinality(S)` - Antalet element i S
- `union(A, B)` - En ny mängd $A \cup B$
- `intersection(A, B)` - En ny mängd $A \cap B$
- `fromList(l)` - En mängd som innehåller elementen i l .

Fundera på hur funktionerna kan använda varandra.

3.

I ett universitetssystem representeras studenter med sina personnummer (som heltal). En deltagarlista för en kurs blir alltså en lista av heltal. Skriv en funktion `percentage` som givet en lista av deltagarlistor beräknar andelen studenter (i procent) som läser alla kurser. Antag att alla studenter läser minst en kurs i listan. Använd `set`!

4.

Ändra implementationen av `set` så att den använder ett binärt sökträd istället för en lista. Antag att du har tillgång till följande datatyp

```
datatype 'a bstree = Void | Bst of 'a * 'a bstree * 'a bstree
```

med tillhörande funktioner:

- `insert(e, T): 'a * 'a bstree -> 'a bstree`
- `delete(e, T): 'a * 'a bstree -> 'a bstree`
- `exists(e, T): 'a * 'a bstree -> bool`
- `size(T): 'a bstree -> int`

Vilka ändringar behöver man göra i programmet i uppgift tre?

Lycka till!