Heapar

Tema: Heapar, implementering av prioritetsköer.

U 1. Visa hur heap ser ut efter insättning av element med följande nycklar: 2, 5, 1, 7, 9, 6, 3, 0, 8, 4.

U 2. En heap implementeras ofta med hjälp av en vektor. Beskriv hur. Visa med en figur hur vektorn som motsvarar heapen från uppgift U 1 ser ut.

U 3. Beskriv kortfattat hur borttagning av minsta elementet i en heap (`poll`) går till.

U 4. I en prioritetskö kan det finnas flera element med samma prioritet. Ibland vill man, vid lika prioritet mellan flera element, vara säker på att elementen kommer att hämtas ut ur kön i den ordning de sattes in. Annorlunda uttryckt: vid lika prioritet vill man att det element som väntat längst tas ut först. Implementeringar av en prioritetskö som uppfyller detta villkor brukar kallas *stabila*.

   a) Visa att heapen inte är en stabil implementering av en prioritetskö. Gör detta genom att, med ett enkelt exempel, visa att lika element inte tas ut i den ordning de satts in.
   b) Finns det något användare kan göra som gör att elementen kommer ut ur heapen enligt ovan?

U 5. Går det bra att använda en heap om vi ofta vill söka efter ett godtyckligt element?

U 6. Vektorn är en lämplig representation av en binär heap eftersom en sådan är ”fylld” på alla nivåer utom möjligen den som ligger längst bort från roten. På denna sista nivå ligger dessutom noderna samlade längst till vänster. Vektorrepresentationen av en binär heap med n noder kan därför utnyttja platserna 0..n-1. Roten placeras på plats 0 och barnen till nod på plats i finns på platserna $2i + 1$ och $2i + 2$. Föräldern till noden på plats i finns på plats $(i - 1)/2$. Vektorn är inte lika lämplig som representation av binära träd i allmänhet. Tänk efter hur stor vektor som i värsta fall skulle krävas för att representera ett binärt träd med n noder i följande två fall

   - trädet är skevt, dvs. har maximal höjd
   - trädet är fyllt av noder på nivåerna 1..k och har dessutom en nod på nivå k+1

Andra sätt att implementera prioritetsköer

U 7. En prioritetskö implementeras ofta med en heap. Diskutera andra sätt att implementera en prioritetskö. Fördelar, nackdelar?

U 8. För specialfallet att element har prioriteter som är heltal i ett visst intervall [i..j] så kan man implementera en prioritetsköklass där alla operationer har tidskomplexitet O(1) och där dessutom prioritetskön är stabil. Förklara hur.