

Ägande och algoritmer

Peter Boström

8 maj 2009

Algoritmer

Det finns nästan ett oändligt antal algoritmer som i sig beskriver väldigt många metoder att utföra väldigt många olika uppgifter. Vissa är effektivare än andra, speciellt i vissa lägen.

Quicksort och andra $O(n \log(n))$ -algoritmer är dramatiskt mycket effektivare vad gäller sortering av större listor. Detta medan Insertion-sort, som knappt är användbar på större listor, är betydligt snabbare på små listor.

Det finns även en uppsjö datastrukturer vars användningsområden skiljer sig rejält. Binärträd för bl.a. sortering, trie för sortering av strängar, hash tables för konstant lookup.

Detta är väldigt grovdraget, men tillgängligheten på algoritmer och datastrukturer över lag, likväl utspridningen av dem, har givit upphov till väldigt många nya för andra användningsområden och specialiseringar. Många variationer utvecklas av fristående personer, men bygger i grund och botten på en redan existerande algoritm eller datastruktur.

Algoritmer är väldigt grundläggande byggstenar i program, och valet av rätt algoritmer är grundläggande för att programmet ska kunna genomarbeta den data som krävs inom rimlig tid. Ett exempel på detta är sorteringsalgoritmen Quicksort. En bra implementation av Quicksort klarar av att sortera en lista med en miljon element på under en sekund på en relativt långsam dator. Detta klarar inte äldre sorteringsalgoritmer av på flera minuter, kanske upp emot en timma.

Quicksort och andra lämpliga algoritmer är alltså nödvändiga för att kunna hantera stora datamängder eller utföra vissa uppgifter inom relativt rimlig tid.

Patent och algoritmer

Den betydelse som väldigt många nya algoritmer har kommit att få visar på hur viktigt det är att algoritmer finns tillgängliga. Som tur är har många betydelsefulla publicerats som forskning från både universitet, högskolor och även företag.

Att patentera algoritmer innebär att kraftigt begränsa utspridningen av de algoritmer som skapas. Som det ligger till nu så skapas väldigt många viktiga algoritmer utan att patenteras.

Risken finns självfallet att motivationen att skapa algoritmer är lägre ifall patentering inte är tillåtet. Det verkar däremot till stor del som att den mängd algoritmer som skapas och finns tillgängliga till väldigt stor del bygger på algoritmer som redan existerar. Kunskapen om dessa skulle dessutom vara mycket mindre, och nya algoritmer skulle inte få bygga till lika stor del på existerande algoritmer och tekniker. Att bygga små variationer på existerande algoritmer skulle också bli jobbigt.

Att principmässigt begränsa tillgången på algoritmer skulle minska utbudet och kvaliteten på väldigt många program, i och med att deras byggstenar inte finns tillgängliga. Detta skulle framför allt vara mycket skadligt för småskaliga projekt och hobbyprojekt som inte drivs med vinstsyfte. Algoritmer som tas fram i vetenskapligt syfte skulle också bli begränsade.

Det är inte alls lika farligt att tillåta patentering av en implementation av en algoritm, då det skulle kunna tillåta konkurrens mellan implementationer. En liknelse är att maple t.ex. inte har patent på den matematik som programmet använder.

Idén är att det fortfarande finns ett intresse av att skapa algoritmer. Värdet av de skapade är väldigt mycket större om de finns tillgängliga för så många som möjligt. Mångfalden och variationen av de som finns är mycket mer utbredd utan de komplikationer som patent medför. Detta vägt emot att enskilda intressen vill behålla kontroll över sina egenskapade algoritmer väger bra mycket tyngre.

Att den stora massan får mycket nytta av skapelserna, och har stor frihet kring användandet av dem väger mycket tyngre, och det är möjligt att det finns ett bättre system för ersättning för forskning kring algoritmer.

Patent däremot är en väldigt konstruktivt hämmande åtgärd, och det är viktigt att algoritmer hålls fria.

I den här frågan håller jag alltså både en utilitaristisk syn på problemet. Patentering av algoritmer är fel eftersom det medför bra mycket mer ont än gott, även om det skulle kränka en potentiell upphovsmans frihet.