Algoritmer (datastrukturer) och komplexitet
våren 2008

För högre betyg på mästarprov 1: Sportturneringar

Detta är en frivillig uppgift för den som vill ha högre betyg på mästarprov 1. Om följande uppgift löses och redovisas senast 9 maj får du betyg A, B eller C (beroende på programmets kvalitet med avseende på funktion och effektivitet). Den som ännu inte är godkänd på mästarprov 1 och inte har för avsikt att få mer än godkänt på momentet ska göra ommästarprov 1 istället, se kursens webbsida. Uppgiften löses individuellt eller i par. Vilket betyg din lösning får (A, B eller C) bestäms vid redovisningen som är på extralabbtiderna 8–9 maj. Senare redovisning är inte möjlig.

$n = 2^m$ spelare deltar i en sportturnering. Turneringen sker i omgångar. I varje omgång spelare varje spelar om möjligt en match. Efter varje omgång sker en ny lottning med ledning av tidigare resultat. Matchresultaten bildar en riktad graf (som kan bevisas vara acyklisk). Tävlingen pågår i m omgångar.

Lottningen av matcher kan ske enligt två metoder, *fjärilsturnering* eller *transitiv poängturnering*. Uppgiften är att värdera vilket system som är effektivast.

I fjärilsturneringen spelar varje spelare mot en annan, slumpmässigt vald spelare med samma historia av vinster och förluster, d.v.s. har lika resultat (vinst eller förlust) i alla tidigare omgångar. Mera konkret: Ta en turnering med 32 deltagare. I första omgången har 16 spelare vunnit och har ”historien” V. 16 spelare har förlorat och har historien F. Båda grupperna blandas slumpmässigt och alla spelar inom samma grupp. Efter andra omgången blir det fyra grupper om 8 spelare med historierna VV, VF, FV och FF. Grupperna blandas och spelar inbördes. Efter tredje omgången blir det 8 grupper om 4 spelare.

I den transitiva poängturneringen räknar man efter varje omgång ut antalet spelare som visat sig vara *bättre* (med ledning av vunna matcher och de slutsatser man kan dra från dem), och *sämre* (med ledning av förlorade matcher och de slutsatser som kan dras från dem). En spelare får poäng som definieras med hjälp av antalet känt sämre spelare minus antalet känt bättre spelare. Efter varje omgång sorteras spelarna efter poäng. Ordningen mellan lag med lika poäng är slumpmässig. Man går igenom listan från högsta till lägsta poäng. Om de två översta spelarna inte har spelat förut under turneringen, så får de spela i nästa omgång, och man tar bort dem från listan. Om de har spelat förut, får den översta spelaren stå över och han tas bort från listan. Därefter fortsätter man med de två översta som finns kvar.

För att avgöra vilken metod som är effektivast simuleras en följd av turneringar. Varje spelare har en *dold styrka*, som är ett heltal s; $1 \leq s \leq n$, där 1 är bäst och n är sämst (alla spelare har olika dold styrka). Den dolda styrkan avgör utgången av matcherna (lägre tal vinner över större). Ordningen mellan lag med samma historik respektive samma poäng ges av slumppermutationer.

Resultatet av en turnering värderas genom antalet kanter i det transitiva höljet till grafen av matcher. Detta har räknats ut efter varje omgång i en transitiv poängturnering, men måste beräknas särskilt efter en avslutad fjärilsturnering. Fler kanter innebär att man känner till fler relationer mellan spelare. Se till exempel http://en.wikipedia.org/wiki/Transitive_closure

Man genomför turneringarna för $n = 8, 16, 32, \ldots$. För varje n genomförs lika många fjärilsturneringar som transitiva poängturneringar. Eftersom turneringar med fler deltagare tar längre tid är det inte lämpligt med lika många turneringar för varje n. För varje n och varje omgång redovisas antalet genomförda turneringar och det genomsnittliga antalet kanter i det transitiva höljet till matchgrafen, dividerat med antalet kanter i den fullständiga grafen med n hörn, dels för fjärilsturneringen och dels för den transitiva poängturneringen.

Försök implementera simuleringsalgoritmerna så effektivt som möjligt, så att du kan köra med så stora n som möjligt.

Forskaren Hans Block, som har bidragit med problemet, vill använda de bästa lösningarna av uppgiften i sin forskning.