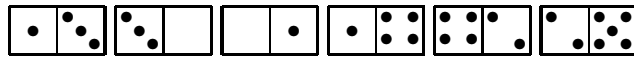


2D1320, TENTAMEN I TILLÄMPAD DATALOGI
Onsdagen den 11 januari 2006 kl 14–19

Maxpoäng = 100. Betygsgränser: 50 poäng ger trea, 70 ger fyra, 90 ger femma. Resultatet anslås om ca två veckor på Nadas anslagstavla på Osquars backe 2, plan 3. Hjälpmedel : En algoritmbok och lila formelbladet.

1. *Dominospel*

En dominobricka består av två kvadrater, vardera med noll till sex prickar på. Ursprungligen representerade varje dominobricka ett av utfallen vid kast med två tärningar (de två kvadraterna representerar alltså två tärningar). Ett roligt tidsfördriv är att ta en näve brickor och försöka lägga ut **alla** dessa i en lång orm, där angränsande kvadrater har lika många prickar.



Du ska nu planera för ett datorprogram som läser in brickorna och sedan anger en lösning om det finns någon.

- (4p) a. Rita först upp problemträdet (roten och några noder i de första två nivåerna räcker).
- (4p) b. Motivera ditt val av algoritm.
- (8p) c. Beskriv hur du skulle implementera algoritmen (arbetsgång, datastrukturer, klasser och metoder).
- (4p) d. Uppskatta komplexiteten för din algoritm!

2. *Dominoträd*

Anta att vi istället lägger ut brickorna i form av ett binärträd, där någon av kvadraterna på barnet måste matcha någon av kvadraterna på föräldern.

- (10p) Formulera en rekursiv tanke för metoden
`printTreeLevel(int n, BinaryNode node)`
som vid anropet
`printTreeLevel(4, root)`
skriver ut alla brickor på nivå 4 i trädet med roten `root`.

3. *Abstrakta dominobrickor*

- (8p) Vilken datatyp är en dominobricka? en tvåsiffrig `int` eller kanske en `int[2]`? Förklara varför ingen av dessa typer är helt idealisk. Beskriv fördelarna med en abstrakt datatyp istället, ange några viktiga metoder och berätta hur man kan använda `interface` och `class` för att implementera datatypen i Java.

4. *Domino-låtar*

- (10p) På skivbolaget Dominos webbplats finns låtar för nerladdning. Varför vill Dominos att låtarna ska vara komprimerade? Varför vill kunderna att låtarna ska vara komprimerade? Och hur komprimerar man musik?

5. *Teori*

(20p) Nedan finns fem frågor om algoritmer och datastrukturer. Varje fråga kan ge upp till fyra poäng. Motivering krävs!

- a. Givet n namn, där n är stort; vilket är snabbast, att sortera namnen eller att skapa alla parvisa kombinationer?
- b. Kommer en postorder-utskrift av noderna i ett balanserat binärträd att ge noderna i omvänd ordning mot en preorder-utskrift?
- c. Kan reguljära uttryck användas för icke-förstörande textkomprimering?
- d. Räcker next-vektorn och sökordet för att definiera en KMP-automat?
- e. Vilken datastruktur passar bäst vid implementationen av en ångra-funktion, en stack eller en kö?

6. *Dominosökning*

(10p) Konstruera och rita upp en KMP-automat som söker efter prickföljden 13311331. Ange även next-vektorn!



7. *Dominoes Inc*

Företaget Dominoes Inc säljer alla jordens spel och leksaker. Dom har 1 073 741 824 olika artiklar i databasen. Hittills har man använt binärträd för att få snabb sökning, men man funderar på att övergå till hashning. Tyvärr har man inte plats för mer än en miljon element i hashvektorn, så man är osäker på om det skulle löna sig att byta.

(10p) Hur snabb skulle hashsökningen bli jämfört med binärsökningen? Du kan anta att antalet jämförelser bestämmer tiden.

8. *Pizzasyntax*

Pizza-kedjan Dominos vill ha ett program som automatiskt tolkar en telefonbeställning. Till din hjälp finns en taligenkännare som kan identifiera alla pizzanamn, heltal, orden *extra* och *utan* samt alla pålägg (ost, tomat, ansjovis mm).
(8p) Skriv en syntax som känner igen en pizzabeställning med obegränsat antal olika pizzor.

(4p) Hitta på två olika exempel på pizzabeställningar och visa hur din syntax fungerar med dessa.