



LUNDS TEKNISKA
HÖGSKOLA
Lunds universitet

Institutionen för
REGLERTEKNIK

FRTN20 – Market Driven Systems

Tentamen 26 maj 2010, 8.00 - 13.00

Poängberäkning och betygssättning

Lösningar och svar till alla uppgifter skall vara klart motiverade. Tentamen omfattar totalt 25 poäng. Poängberäkningen finns markerad vid varje uppgift. Preliminära betygsgränser:

Betyg 3: 12 poäng
4: 17 poäng
5: 22 poäng

Tillåtna hjälpmedel

Tentamensresultat

Resultatet anslås senast måndagen den 31 maj på institutionens anslagstavla på första våningen i Maskinhuset samt på institutionens hemsida. Visning den 31 maj kl 11.30–12.00 på reglerteknikinstitutionen, i M:2171 (Charlotta Johnssons kontor), M-huset våning 2.

Lösningar till tentamen 26/05/10

1.

- a. En snickare tillverkar bord och stolar. Marknadspriset är p_1 kronor per bord och p_2 kronor per stol. Råvaran är trä, och det går åt r_1 plankor per bord och r_2 plankor per stol. En plankor kostar q kronor. Därtill kommer tillverkningstiden som beräknas vara t_1 timmar per bord och t_2 timmar per stol. Att hålla snickeriet öppet kostar s kronor per timme.

Snickaren planerar att upphöra med möbeltillverkning och använda lokalen som bilverkstad. Först vill han dock göra sig av med det befintliga råvarulagret bestående av r plankor. Av schematekniska skäl får detta ta högst t timmar.

Ställ upp ett linjärt program, på kanonisk form, vars lösning maximerar förtjänsten. (1.5 p)

- b. Efter att ha identifierat samtliga parametrar, står snickaren inför följande problem:

$$\begin{aligned} \max_{x_1, x_2} \quad & \begin{bmatrix} 2000 & 400 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ \text{subject to} \quad & \begin{bmatrix} 20 & 10 \\ 24 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \preceq \begin{bmatrix} 200 \\ 120 \end{bmatrix}, \quad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \succeq \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned}$$

Lös det åt honom. Vad representerar variablerna x_1 och x_2 ? (1.5 p)

- c. Beakta att det endast går att tillverka ett heltal bord respektive stolar. Kommentera hur detta påverkar din lösning från föregående deluppgift. Hur många bord respektive stolar ska snickaren tillverka. (1 p)

Solution

- a. Låt x_1 vara antalet bord som ska tillverkas och x_2 antalet stolar. Den totala förtjänsten, som snickaren önskar maximera, ges av $(p_1 - r_1q - t_1s)x_1 + (p_2 - r_2q - t_2s)x_2$.

Antalet tillverkade möbler får inte vara negativt, dvs $x_1 \geq 0$ och $x_2 \geq 0$. Därtill kommer den begränsade tillgången på plankor $r_1x_1 + r_2x_2 \leq r$ och tid $t_1x_1 + t_2x_2 \leq t$.

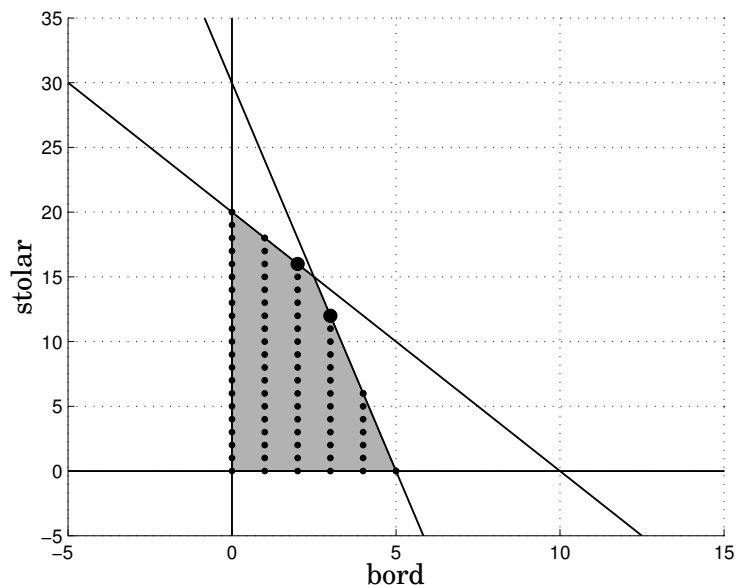
Detta motsvaras av det linjära programmet

$$\begin{aligned} \max_{x_1, x_2} \quad & \begin{bmatrix} (p_1 - r_1 - t_1s) & (p_2 - r_2 - t_2s) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ \text{subject to} \quad & \begin{bmatrix} t_1 & t_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq t, \quad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \succeq \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \end{aligned}$$

- b. Bivillkoren definierar det skuggade området i figur 1.

Hörnen på området är

$$x_a = (0, 0), \quad x_b = (5, 0), \quad x_c = (2.5, 15), \quad x_d = (0, 20)$$



Figur 1 Linjära programmet i snickarens problem.

Utvärdering av målfunktionen:

$$c^T x_a = 0, \quad c^T x_b = 0, \quad c^T x_c = 0, \quad c^T x_d = 0$$

Alltså bör snickaren tillverka 2.5 bord och 15 stolar.

- c. Eftersom snickaren ej kan sälja ett halvt bort, kan man misstänka att materialet kan användas på bättre vis än vad lösningen till förra deluppgiften föreslår. Möjliga heltalslösningar är utritade som prickar i figur 1. Vi ser att de två som är av intresse är

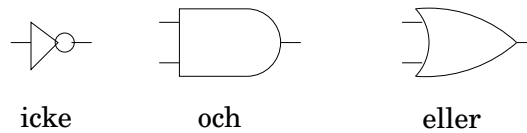
$$x_e = (2, 16), \quad x_f = (3, 12)$$

Utvärdering av målfunktion:

$$c^T x_e = 10400, \quad c^T x_f = 10800$$

Eftersom 10800 kr är mer än målfunktionsvärdet i något av heltalshörnen från föregående deluppgift drar vi slutsatsen att snickaren ska tillverka 3 bord och 12 stolar.

2. I denna övning kommer vi använda logiska grindar för att konstruera ett 1-bits minne (en så kallad SR-vippa). Minnet har två ingångar: set-ingången s och reset-ingången r samt en utgång q . Då $r = 1$ och $s = 0$ sätts $q = 0$. Om istället $r = 0$ och $s = 1$ sätts utgången till $q = 1$. När $r = s = 0$ bevaras värdet på utgången. Insignalkombinationen $r = s = 1$ är förbjuden.
 - a. Beteckna med q^+ utgångens kommande värde som följd av nuvarande värde q samt insignalerna r och s . Rita en sanningstabell som relaterar q^+ till r, s och q . (1 p)
 - b. Vi har $q^+ = f(q, r, s)$. Skriv ner ett booleska uttrycket för f . (1 p)



Figur 2 Logiska grindar.

- c. Rita ett grindnät som realiserar SR-vippan. Grindarna du får använda finns återgivna i figur 2. (1 p)

Solution

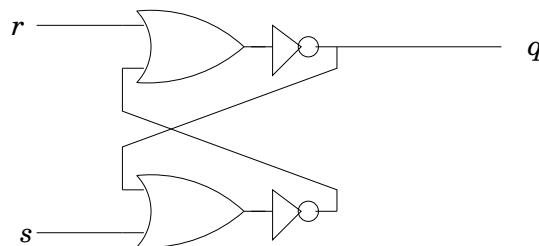
a.

r	s	q	q^+
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	X
1	1	1	X

- b. Utifrån sanningstabellen kan man direkt skriva ner (till exempel)

$$q^+ = \bar{r} \cdot q + s$$

- c. En realisering är den i figur 3.



Figur 3 Grindrealisation av SR-vippa.

3.

- a. I kursen har vi använt flera sätt att beskriva sekvenser, däribland

- Petrinät
- Grafcet
- JGrafchart

Förklara kortfattat hur dessa verktyg hänger samman och vad skillnaden mellan dem är. (1 p)

- b.** Använd JGrafchart för att skriva/rita ett program för automatiserad kak-tillverkning i en fabrik, beskrivet nedan.

Utrustningen består av en ugn. Variabeln *heat*, med omfång 0-100, anger hur mycket ström som skall drivas genom ugnens värmeslingor. Variabeln *temp* anger ugnens temperatur i Celcius. En robot stoppar in en bakplåt i ugnen och stänger dörren. Detta indikeras av att variablerna *door* och *tray* får värden skilda från 0. När detta sker ska variabeln *ready* sättas till 0, varpå ett macro-steg för en P-regulator ska köras. Efter att macro-steget avslutats ska exekveringen återgå till starttillståndet och *ready* sättas till 1.

Innan en ny batch kan gräddas måste programmet kontrollera att dörren öppnats, plåten tagits ut, en ny plåt stoppats in samt att dörren stängts igen. (Man utgår från att det finns en tom plåt i ugnen och att dörren är stängd, då programmet startas.) (1 p)

- c.** Implementera macrosteget för P-regulatorn. Det ska exekvera tills dess att *timer* har ett värde större än 40. (Timern startas automatiskt när *ready* sätts till 0.) Referenstemperaturen för gräddning är 225 °C. Regulatorn kör inte med en fix sampelhastighet, utan exekverar istället så fort det går. Regulatorns förstärkning ska vara 5. Tänk på att styrsignalen ej får överskrida 100 eller underskrida 0. (1 p)

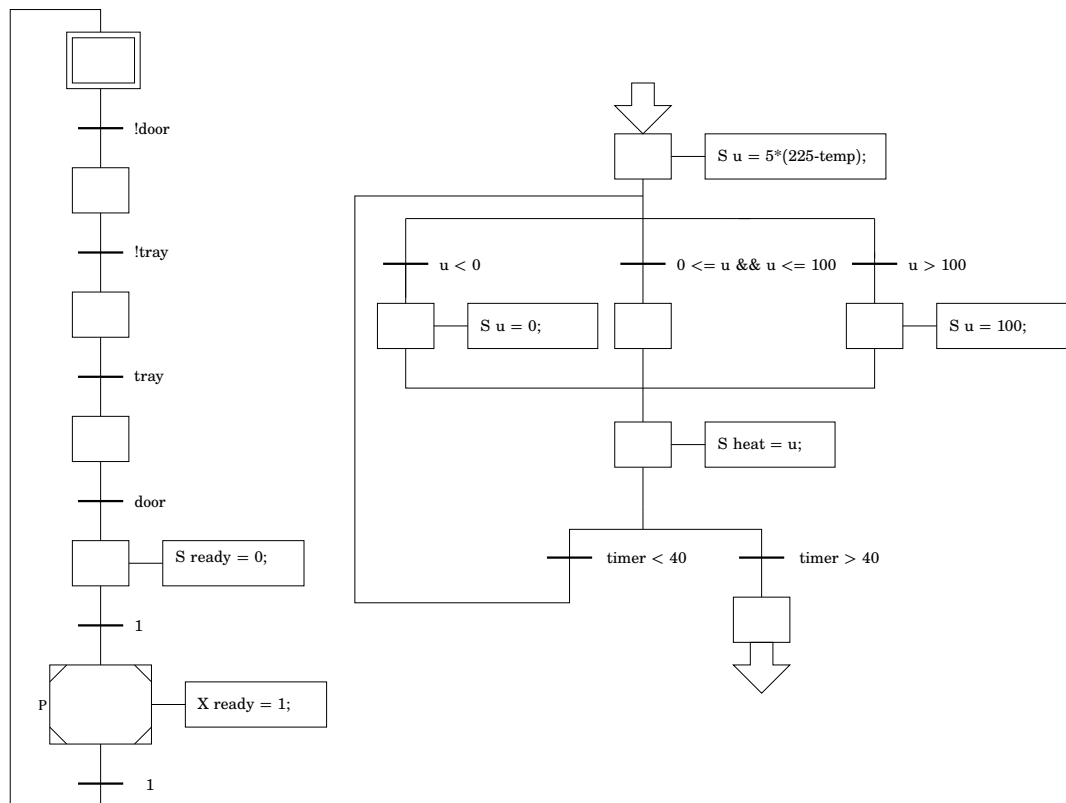
Solution

- a.** Petrinät är ett teoretiskt modellerings- och simuleringsverktyg. Det finns en rik bakomliggande teori, varför Petrinät kan användas i avancerande analyser.

Grafcet är ett språk som bygger på samma grund som Petrinätet (tokens, platser och transitioner). Grafcet är emellertid mer applikationsinriktat och har variabler och kontrollstrukturer som Petrinätet saknar.

JGrafchart är en implementation av Grafcet. Det är alltså ett verktyg för att skriva program som bygger på Grafcet och sedan kan exekveras på riktig hårdvara.

- b.** Se nästa delproblem.
- c.** I figur 4 visas ett lösningsförslag. Huvudprogrammet till vänster i bilden, macrostegets innehåll till höger.



Figur 4

4.

a. Förklara korrekt och kortfattat vad följande (engelska) attribut innebär för ett Petrinät.

- live
- deadlock-free
- bounded

(1.5 p)

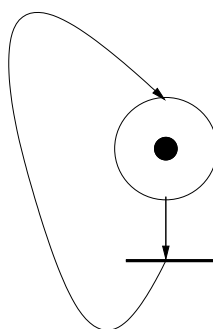
b. Rita ett Petrinät som är live, deadlock-free och bounded.

(1 p)

Solution

- a.**
- I ett live-nät kan inga transitioner bli unfirable.
 - I ett deadlock-fritt nät finns alltid åtminstone en transition som kan avfyras.
 - Ett bounded nät har, vid varje tidpunkt, ett begränsat antal tokens.

b. I figur 5 ges exempel på ett nät som har samtliga attribut.



Figur 5 Petrinätexempel.