

Några övningar på logiska uttryck

Jag har valt allt formulera övningarna så att svaren utgörs av punktmängder, övningarna ger då samtidigt lite träning på flervariabelanalys.

I följande övningar vill man veta för vilka tal x och y uttrycken är sanna? Svara med en geometrisk beskrivning (t.ex. rektangeln med hörn i ... etc) eller rita en bild på papper.

Lösningförslag hittar du här: `/chalmers/groups/thomas_math/MATL/Ovningar/villkor.m`
Under Windows: `\\sol.ita.chalmers.se\groups\thomas_mat\MATL\Ovningar\villkor.m`

Programmet `villkor.m` ritar upp en approximation av punktmängden (approximation eftersom programmet utnyttjar **ändligt** många punkter). I några fall är inte mängden begränsad varför endast en del av punktmängden är utritad.

För att köra lösningen (under Linux) kopiera först filen till aktuella katalog (glöm inte sista punkten):

```
cp /chalmers/groups/thomas_math/MATL/Ovningar/villkor.m .
```

Kör sedan programmet genom att skriva `villkor` i MATLAB, och följ därefter instruktionen som kommer upp. Du kan läsa koden i `villkor`, men den kan framstå som obegriplig. Du kommer inte att få lära dig tillräckligt mycket MATLAB, i den här kursen, för att första programmet i alla detaljer.

1. $\text{abs}(x) \leq 1 \ \& \ \text{abs}(y) \leq 2$
2. $\text{abs}(x) \leq 1 \ | \ \text{abs}(y) \leq 2$
3. $\sim(\text{abs}(x) \leq 1) \ \& \ \text{abs}(y) \leq 2$
4. $\sim\text{abs}(x) \leq 1 \ \& \ \text{abs}(y) \leq 2$
5. $\sim(\text{abs}(x) \leq 1) \ \& \ \sim(\text{abs}(y) \leq 2)$
6. $\text{abs}(x) > 1 \ \& \ \text{abs}(y) > 2$
7. $\sim(\text{abs}(x) \leq 1 \ | \ \text{abs}(y) \leq 2)$
8. $\sim(\text{abs}(x) \leq 1) \ | \ \sim(\text{abs}(y) \leq 2)$
9. $\text{abs}(x) > 1 \ | \ \text{abs}(y) > 2$
10. $\sim(\text{abs}(x) \leq 1 \ \& \ \text{abs}(y) \leq 2)$
11. $\text{abs}(x) \leq 1 \ \& \ \text{abs}(y) \leq 2 \ | \ x + y \leq 3$
12. $x \leq 1 \ \& \ 0 \leq x \ | \ y \leq 3 \ \& \ 2 \leq y \ \& \ \text{abs}(x + y) \leq 4$
13. $\text{abs}(x) \leq y$
14. $\text{abs}(x) \leq y^2$
15. $x * y \leq 1$

Några övningar på loopar

Lös följande övningar på det mest uppenbara sättet. Senare i kursen kommer vi att kunna göra kortare och effektivare lösningar med hjälp av vektorer. Lösningförslag hittar du här (i en enda lång MATLAB-fil):
 /chalmers/groups/thomas_math/MATL/Ovningar/loopar.m .

Under Windows: \\sol.it.chalmers.se\groups\thomas_mat\MATL\Ovningar\loopar.m .

1. Gör en loop som skriver ut de udda talen 1, 3, 5, ..., 99.
2. Gör en loop som skriver ut de jämna talen 100, 98, 96, ..., 2.
3. Gör en loop som skriver ut talen -1, 2, -3, 4, ..., -99, 100.
4. Gör en loop som skriver ut produkterna 1 · 2 · 3, 2 · 3 · 4, ..., 98 · 99 · 100.
5. Gör en loop som skriver ut de 20 första Fibonaccitalen, dvs. talen

$$1, 1, 2, 3, 5, 8, 13, \dots$$

dvs. nästa tal är summan av de två föregående (och vi startar med 1, 1).

6. Skriv ett MATLAB-program som beräknar och skriver ut det minsta n för vilket gäller att

$$\sum_{k=1}^n \frac{1}{k^2} \geq 1.6$$

7. Skriv ett MATLAB-program som beräknar och skriver ut alla n för vilket gäller att

$$1.6 \leq \sum_{k=1}^n \frac{1}{k^2} \leq 1.62$$

8. Låt $s_N = 1 + 1/2^4 + 1/3^4 + \dots + 1/N^4$. Beräkna, i Matlab, det minsta N för vilket gäller att: $s_N = s_{N+1}$, när summorna beräknas från vänster till höger. Vilket är värdet på s_N för detta N ?

Ovanstående är orimligt ur matematisk synpunkt, men med flyttalsaritmetik (är inte Matlabs "fel", med andra ord) kommer man att råka ut för fullständig utskiftning (pga att man räknar med ett ändligt antal decimaler). Fullständig utskiftning inträffar när "stort tal + litet tal = stort tal", dvs. inget av det lilla talet kommer med. T.ex. gäller att

```
>> (1 + 1e-16) - 1
ans = 0
```

9. Skriv ett MATLAB-program som skriver ut maximum (och motsvarande värde på k) av $\sin k$, $k = 1, 2, 3, 4, \dots, 100$.
10. Skriv ett MATLAB-program som skriver ut minsta värdet på $e^{1-x} \sin y$ där $x = 0.1, 0.2, \dots, 1.0$ och $y = 0.1, 0.2, \dots, 1.0$. Även motsvarande x och y skall skrivas ut.
11. Använd två loopar för att generera en utskrift i stil med:

```
1 1
1 10
1 1000
1 10000
2 1
2 10
```

```
2 1000
2 10000
osv.
9 1
9 10
9 1000
9 10000
```

12. Använd tre loopar för att generera en utskrift i stil med:

```
-5 1 10
-5 1 100
-5 1 1000
-5 2 10
-5 2 100
-5 2 1000
-5 3 10
-5 3 100
-5 3 1000
-4 1 10
-4 1 100
-4 1 1000
-4 2 10
-4 2 100
-4 2 1000
-4 3 10
-4 3 100
-4 3 1000
-3 1 10
-3 1 100
-3 1 1000
-3 2 10
-3 2 100
-3 2 1000
-3 3 10
-3 3 100
-3 3 1000
```

13. Skriv ett MATLAB-program som skriver ut minsta värdet på $ze^x \sin 5y$ där $x = 0.1, 0.2, \dots, 1.0$, $y = 0.1, 0.2, \dots, 1.0$ och $z = 0.1, 0.2, \dots, 1.0$. Även motsvarande x , y och z skall skrivas ut.

14. Beräkna det minsta n för vilket gäller att:

$$\sum_{k=1}^n \left(\sum_{j=k}^n \frac{1}{j+k} \right) > 8$$

Vilket är värdet på summan för detta n ?

15. En Pythagoreisk trippel är tre positiva heltal p, q, r sådana att $p^2 + q^2 = r^2$, t.ex. tripplarna 3, 4, 5 och 5, 12, 13. Skriv ett MATLAB-program som hittar alla tripplar med $1 \leq p, q, r \leq 100$. Hur många olika sådana tripplar finns det? Notera att 3, 4, 5 är samma trippel som 4, 3, 5.

Några övningar på funktioner

Övningarna är till för träning, jag påstår inte att det är speciellt användbara funktioner. I flera fall finns det dessutom färdiga MATLAB-funktioner man skulle kunnat använda. Jag har valt att **inte** använda dessa i mina lösningar! Man kan vektorisera en del funktioner, koden blir då kortare och effektivare. Jag har dock avstått från vektorisering i mina lösningsförslag. Jag preallokerar inte heller vektorer och matriser i mina funktioner, något man bör göra i praktiken. Skriv också huvudprogram som anropar dina funktioner. Lösningsförslag hittar du här (i en enda lång MATLAB-fil): `/chalmers/groups/thomas_math/MATL/Ovningar/funktioner.m`. Under Windows: `\\sol.it.chalmers.se\groups\thomas_mat\MATL\Ovningar\funktioner.m`.

1. Skriv en funktion `max2(x, y)` som tar två tal `x` och `y` som inparametrar och som returnerar det största värdet.
2. Skriv en funktion `[first, middle, last] = fml(x)` som givet vektorn `x` returnerar första, det mittersta och det sista elementet i vektorn. Om `x` inte har ett udda antal element skall `middle` sättas till medelvärdet av de två mittersta elementen. Du kan anta att `x` innehåller åtminstone ett element (vektorn är inte tom, med andra ord).
3. Skriv en funktion `[xpry, xmry] = vec_sums(x, y)` som givet vektorerna `x` och `y` returnerar `xpr2 = x+rev(y)` samt `xmr2 = x-rev(y)`, där `rev(y)` betecknar den vektor man får om man kastar om elementen i `y`, så `rev([3 1 7])` är `[7 1 3]`. Observera att `rev` inte finns färdig i MATLAB. Du kan anta att `x` och `y` har samma antal element och samma orientering (båda är rader eller båda är kolonner).
4. Skriv en funktion `z = f(x, y)` som beräknar `z = sin(x)*exp(x) + sin(y)*exp(y)`. För att träna på funktioner, skriv en funktion `w = g(v)` som beräknar `w = sin(v)*exp(v)` och anropa `g` från `f`. Man skulle normalt inte gör på detta sätt när man har så enkla uttryck
5. Skriv en funktion `P = positive(M)` som givet en matris `M`, returnerar en logisk matris `P` sådan att `P(j, k)` har värdet `true` då `M(j, k)` är positivt. För övriga värden skall `P(j, k)` ha värdet `false`.
6. Skriv en funktion `crypt = secret(text, shift)` som tar en sträng `text` och producerar den krypterade strängen `crypt` genom att använda ett sk Caesarchiffer. Antag att `text` innehåller en mening på engelska skriven med gemena (små bokstäver). I ett Caesarchiffer ersätts en bokstav i strängen `text` med den bokstav som kommer `shift` positioner (ett heltal) senare i alfabetet (eller tidigare i alfabetet om `shift` är negativt). Om man hamnar "utanför" alfabetet börjar man från början (eller slutet), ungefär som man hade skrivit alfabetet på en cirkel i stället för på en rad. Skiljetecken krypteras inte. Så här ser alfabetet ut i MATLAB: `abcdefghijklmnopqrstuvwxyz`.

Exempel: Om `text = the money is in the third book` och `shift = 10` får vi
`crypt = dro wyxoi sc sx dro drsbn lyu`.
 Låt oss se på hur bokstäverna `e` och `t` transformeras.

1	2	
12345678901234567890123456	nummer	
abcdefghijklmnopqrstuvwxyz		

Så `e` har position 5, $5+10 = 15$ som svarar mot `o` som är den nya bokstaven. `t` har position 20, $20+10 = 30$, subtrahera 26 (antalet bokstäver i alfabetet) så får vi $30 - 26 = 4$, så bokstaven blir `d`. Omvänt, om `shift` hade varit -10 så hade `o` blivit `e` och `d` hade blivit `t`. Vi kan alltså gå från `crypt` till `text` genom att anropa `secret` med ombytt tecken på `shift`.

Ledning: MATLAB-funktionen `mod` är användbar i detta sammanhang.