NADA, KTH **Betygsgradering:** Grunduppgift medel, betyg E-A
DD1310/DD1314 våren 2013 (Python)

174 Patiensen Klockan

Läs anvisningar och betygsregler på kurshemsidan!!!

Varudeklaration: Datastrukturer, grafik (i första extrauppgiften).

Patiensen Klockan går till så här: Man blandar en vanlig kortlek med 52 kort. Sedan lägger man ut de tretton översta korten i leken i en rad på bordet. Alla kort läggs med framsidan upp, så att man kan se deras valörer.

Nu gäller det att få ett ess överst i den första högen, en tvåa överst i den andra o s v. I elfte högen ska en knekt ligga, i tolfte en dam och i trettonde en kung. Till sin hjälp har man resten av kortleken. Man fortsätter att lägga kort i samma ordning som förut med skillnaden att om rätt valör redan ligger på en plats hoppas denna över. De nya korten täcker delvis de gamla. Patiensen går ut om man lyckas få rätt valör överallt, och misslyckas om högen tar slut utan att patiensen gått ut.

Du ska göra ett program som blandar en kortlek och lägger Klockan. Visa en rad på skärmen för varje varv du lägger i Klockan. En omgång (som inte gick ut) kan se ut så här:

```
KLOCKAN:

K2  R9  S5  K6  KD  RT K3  KK  HK  HD  R2  S6  RK
SD  H9  RJ  ST  H3  SK R7  HT  K8  S2  H6  R8
SA  HA  R3  H7  HJ  H4     S3  R4  K7  K9  RD
    K5      S4  S8  S7     S9  SJ  H5  H2
    H8          RA  KA     K4  R5  R6  KT
    KJ

Högen tog slut utan att patiensen gick ut. Låt knekt representeras av
   bokstaven J, som i engelska ''jackets'' och 10 med bokstaven T.
```

Här ser man att ruter kung hamnade på rätt plats redan i utlägget. Sedan lades inga fler kort på plats tretton. Ruter sju hamnade rätt på andra utlägget osv. Högen tog slut med klöver knekt. Därför är inga fler kort lagda i sjätte raden. Programmet kommenterar resultatet.

Ditt program ska låta användaren välja mellan:

- Se utlägg av en patiens.
- Se statistik på valt antal patienser (dvs användaren anger ett antal, programmet lägger så många patienser utan att visa utlägg på skärmen och berättar sedan hur många som gick ut)
- Sluta (q)

Extrauppgift, betyg C: Inför felhantering. Om användaren matar in ett felaktigt val ska programmet meddela detta och be om en ny inmatning. Om användaren väljer statistikvalet ska programmet kontrollera att det är ett heltal som användaren matar in, om så inte är fallet ska programmet be om ett nytt värde.

Extrauppgift, betyg B: Du har säkert upptäckt att Klockan inte går ut speciellt ofta. Modifiera ditt program så att när ett kort hamnar på rätt plats på bordet, hamnar de felaktiga kort som låg där innan sist i högen av olagda kort. Dvs när Ruter sju läggs i exemplet ovan, hamnar klöver tre sist i högen av olagda kort. När spader ess läggs, hamnar klöver två näst sist och spader dam sist i högen. (OBS ordningen!) De kort som hamnar sist i högen läggs på så vis ut en gång till på bordet. Egentligen är detta den riktiga versionen av Klockan. Jämför statistikresultaten på den här versionen med resultaten på den gamla!

Extrauppgift, betyg A: Visa hur klockan ser ut när man lägger den (textutmatning duger bra här).

När man lägger Klockan med riktiga kort, brukar man i allmänhet lägga korten i form av en urtavla (därav namnet). Esset hamnar då förstås på ettans plats, tvåan på tvåans och så vidare runt till damen på tolvans plats. Kungen får ligga mitt i urtavlan. Inför grafik i ditt program så att utskriften av patiensen blir i form av klockor i stället. Man vill ju fortfarande kunna se alla steg i patiensen, så en klocka per varv måste visas. Ej nylagda kort i varje varv ska vara med, men man ska tydligt se vilka som är nylagda och vilka som inte är det. Exemplet ovan kan då se ut så här:

```
KLOCKAN:

       S6                  R8                  RD
    R2    K2              H6    SD              K9    SA
  HD        R9          S2        H9          K7        HA
 HK    RK    S5        K8   (RK)   RJ        R4   (RK)   R3
  KK        K6          HT        ST          S3        H7
    K3    KD              R7    H3              (R7)    HJ
       RT                  SK                  H4

      (RD)                (RD)                (RD)
    H2   (SA)            KT   (SA)            (KT) (SA)
  H5        K5          R6        H8          (R6)       KJ
 SJ   (RK)  (R3)       R5   (RK)  (R3)       (R5)  (RK)  (R3)
  S9        S4          K4       (S4)        (K4)      (S4)
   (R7)   S8              (R7)   RA              (R7)  (RA)
       S7                  KA                  (KA)

Leken tog slut utan att patiensen gick ut.
```