

# 119 Ludvig XIVs patiens

Läs anvisningar och betygsregler på kurshemsidan!!!

**Varudeklaration:** Datastrukturer. Rekursion.

Blanda leken och lägg successivt ut korten med framsidan upp bredvid varandra från vänster till höger! Om ett kort har samma färg (sp, hj, ru, kl) eller samma valör som sin närmaste granne till vänster eller som det kort som ligger tre steg till vänster får det läggas ovanpå detta. När ett kort lagts på ett annat flyttas de i fortsättningen tillsammans. Högen betraktas alltså som ett enda kort. När alla kort lagts ut räknar man högarna. Det går ut på att få så få högar som möjligt. Ludvig XIV lade patienten hela livet utan att komma ner till en hög... Ditt program ska simulera patientsläggningen och fråga om vilken hög kortet ska flyttas till när det finns flera högar att välja.

Så här kan det se ut på skärmen:

```
sp 10
-----
sp 10 hj E
-----
sp 10 hj E hj K
-----
sp 10 hj K
-----
sp 10 hj K ru 5
-----
sp 10 hj K ru 5 sp 4
sp 4 hj K ru 5
-----
sp 4 hj K ru 5 sp 5
```

Vilken hög ska kortet sp5 flyttas till? 1

```
sp 5 hj K ru 5
ru 5 hj K
osv...
```

Programmet slutar med att skriva ut hur många högar det blev kvar.

**Extrauppgift, betyg C:** Gör en meny där du låter användaren välja mellan:

- Se utlägg av en patient.
- Se statistik på valt antal patienter (dvs användaren anger ett antal, programmet lägger så många patienter utan att visa utlägg på skärmen och visar statistik över antal högar som blev kvar).

**Extrauppgift, betyg B:** Gör ett program som utför flyttningar och lägger en hel patient. Prova själv spelet med en kortlek för att komma på en algoritm.

**Extrauppgift, betyg A:** Här ska användaren få lägga patienten själv. Leta upp små bilder av kort eller skapa egna bilder och gör gränsnittet grafiskt. Lägg vidare till en knapp där man klickar då man vill stega sig fram genom utlägg.

**Datafiler och hjälpfiler:** [www.csc.kth.se/DD1311/P/random.txt](http://www.csc.kth.se/DD1311/P/random.txt)