134 Springarens vandring på schackbrädet

P-uppgiften ska göras individuellt. Läs CSC:s hederskodex innan du börjar!

Varudeklaration: Datastrukturer, randvillkor.

Skriv ett program som följer en springares vandring över ett 64 rutors schackbräde! Springaren följer schackreglerna, dvs den förflyttar sig antingen två steg horisontellt och ett steg vertikalt eller två steg vertikalt och ett steg horisontellt. Under hela vandringen får springaren inte komma till en plats mer än en gång.

Användaren får ange vilken ruta springaren ska starta på. Det finns högst 8 stycken platser som springaren kan gå till från varje ruta. Du skriver lämpligen en klass vars fält representerar schackbrädet och vars metoder tex. förflyttar springaren. Står vi på en kantruta kan vi inte flytta springaren i vissa riktningar, ty då hamnar vi utanför brädet. I tipset finner du en variant att lösa detta problem. Programmet ska plocka ut de godkända alternativen och sedan med hjälp av slumpen välja vilken ruta som springaren ska gå till.

Exempel:

```
nyPlats = random.randrange(antalOKplatser)
```

Du får numrera de godkända platserna enligt en fast ordning, tex medsols. Programmet tar slut då springaren inte har någon plats att flytta till. Användaren ska starta programmet genom att ange en startpunkt på schackbrädet. Därefter ska brädet skrivas ut. I presentationen av schackbrädet ska det tydligt framgå:

- Rutnätet, varje koordinat har en egen ruta.
- Springarens förflyttningar ska vara markerade, en ruta kan då innehålla:

  Tomrum , springaren har inte besökt denna ruta.

  Ett tal , springaren har besökt denna ruta, i steg nr ...

- Koordinater i kanterna.

Tips: Schackbrädet kan lämpligen representeras av en matris som förutom det verkliga schackbrädet (rad 2 – 9, kolumn 2 – 9) har ytterligare två rader och kolumner runt omkring (som flyttbuffert). Matrisen får då indexgränserna (0..11). På raderna 0 och 1 såväl som i kolumnerna 10 och 11 sätts alla element till -1 för att markera att rutorna ligger utanför brädet. För att markera rutornas tillstånd ska du använda dig av någon konvention.

T ex kan olika tal ge information om rutan:

-1 Rutan ligger utanför schackbrädet.

0 Rutan ligger på schackbrädet, springaren har aldrig varit här.

k Rutan ligger på schackbrädet, springaren har varit här tidigare, i steg nummer k (där k är ett tal > 0).

Utskriften från programmet kan se ut så här:

```
Springarens vandring:
   -----------------------------------------
   |   --------------------------------   |
   |8  |   |   |   | 2 |   |   |   |   |   |
   |   --------------------------------   |
   |7  |   |   |   |   |   | 3 |   |   |   |
   |   --------------------------------   |
   |6  |   |   | 1 |   |   |   |   |   |   |
   |   --------------------------------   |
   |5  |   |   |   |   |   |   |   |   |   |
   |   --------------------------------   |
   |4  |   |   |   |   |   |   |   |   |   |
   |   --------------------------------   |
   |3  |   |   |   |   |   |   |   |   |   |
   |   --------------------------------   |
   |2  |   |   |   |   |   |   |   |   |   |
   |   --------------------------------   |
   |1  |   |   |   |   |   |   |   |   |   |
   |   --------------------------------   |
   |     A   B   C   D   E   F   G   H     |
   -----------------------------------------
```

Extrauppgift, betyg C: Ge användaren möjlighet att mata in en egen springarvandring. Kontrollera att inmatade koordinater motsvarar en tillåten ruta.

Extrauppgift, betyg B: Programmet ska uppdatera en "high-score-fil" mellan körningarna. Textfilen läses in vid programstart. Om körningen resulterade i fler steg än det hittills största, som är lagrat i textfilen, ska det nya antalet steg lagras på textfilen.

Extrauppgift, betyg A: Ge programmet en snyggare presentation genom att visa resultatet i ett separat grafikfönster. Låt användaren få se ett steg i taget.