NADA, KTH **Betygsgradering:** D-A
2D1312 2006-2007 (python)

169 Tärningsspel

Läs anvisningarna på kurshemsidan!!!

Varudeklaration: Slumptal, datastrukturer, tidtagning och matematiska funktioner (första extrauppgiften), filhantering (andra extrauppgiften), grafiskt användargränssnitt (tredje extrauppgiften)

Tänk dig följande spel: En spelare har sju tärningar: en röd, en blå och fem vita. Han slår först den röda och den blå tärningen och skapar ett tvåsiffrigt tal mål enligt

$$\text{mål} = 10 \cdot (\text{det den blå tärningen visar}) + (\text{det den röda tärningen visar})$$

Sedan slår han de fem vita tärningarna, och hans uppgift är att kombinera dessa fem med hjälp av de fyra räknesätten för att få det tvåsiffriga talet. Du ska skapa ett program som simulerar tärningskasten, låter användaren spela och kontrollerar hans resultat. Så här kan det se ut:

```
Jag slår tärningarna:
blå = 2
röd = 5
vita = 2 4 4 1 3

Du ska alltså skapa talet 25 av talen 2, 4, 4, 1 och 3
med hjälp av de fyra räknesätten.

Din uträkning: 4+3*4-1-2
BRAVO! Du klarade det!

Vill du spela igen? (j/n)
```

Användaren ska alltså få skriva in hela sin uträkning på en rad. Programmet behöver inte ta hänsyn till operationernas prioritet utan evaluerar från vänster till höger. Det innebär att uttrycket ovan evalueras som

$$4 + 3 \cdot 4 - 1 - 2 = ((((4 + 3) \cdot 4) - 1) - 2)$$

Givetvis ska användaren informeras om detta i inledningstexten till körningen. Varje tärning ska användas exakt en gång.

Tips: Istället för att lagra varje vit tärning kan du lagra antalet ettor, antalet tvåor etc så blir det lättare att hålla reda på vilka vita tärningar som använts.

VGV

Extrauppgift, betyg B: För att göra spelet intressantare ska du poängsätta användarens lösning. Poängen ska bero av två saker: Hur bra lösningen är och hur fort han får fram denna lösning. Man kan skriva uttrycket för poängen som

$$\text{poäng} = k \cdot \text{tidspoäng} \cdot \text{kvalitetspoäng}$$

där

- k är en konstant
- Tidspoäng väljs som någon funktion av tiden t som går mot oändligheten då t går mot noll och mot noll då t går mot oändligheten. Exempel på sådana funktioner är $1/t$, $1/\sqrt{t}$, $1/t^2$.
- Kvalitetspoäng är ett mått på hur bra lösning användaren presterat. Denna poäng ska ha tre diskreta nivåer:
  - Bäst ($\Rightarrow$ högst kvalitetspoäng) är lösningen om användaren lyckats använda varje vit tärning exakt en gång.
  - Halvbra ($\Rightarrow$ halvhög kvalitetspoäng) är att lyckas använda varje vit tärning högst en gång, men att inte få med alla fem i uträkningen.
  - Sämst ($\Rightarrow$ lägst kvalitetspoäng) är lösningen om man använder minst en vit tärning mer än en gång.

Testa din poängsättning! Se till att det lönar sig att göra en bättre lösning, som tar längre tid att fundera ut, i stället för en sämre på kort tid! Se också till att poängen blir så hög att du kan avrunda den till ett heltal. En felaktig lösning ger förstås inga poäng alls, utan bara ett beklagande:

```
ATTANS OSIS! Din uträkning blev 23. Det skulle ju bli 25!
```

Tips: Läs i filen `tidochdatum` för att komma på hur du ska mäta användarens tid!

Extrauppgift, betyg B: Gör en tio-i-topp-lista med namn och poäng på de tio bästa spelarna i poängordning (bästa spelaren överst). Listan ska lagras i en fil mellan körningarna!

Extrauppgift, betyg A: Gör ett grafiskt användargränssnitt som låter användaren klicka på siffror och operatorer för att skapa talet. Både det beräknade uttrycket och det värde man beräknat ska visas under tiden. Dessutom ska det finnas en ÅNGRA-knapp som för varje tryckning backar ett steg i uttrycket.