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

133 Sänka skepp

Läs anvisningarna på kurshemsidan!!!

Varudeklaration: Datastrukturer. Viss skärmhantering. I extrauppgiften tillkommer sortering.

Du ska skriva ett program som slumpvis lägger ut en mängd fartyg på en yta. Användaren ska sedan kunna beskjuta ytan och programmet ska meddela resultatet (envägs beskjutning). Spelet avslutas då användaren så önskar eller efter att alla fartyg är sänkta. Regler för utläggning av fartyg:

- Två fartyg får aldrig ligga precis intill varandra, utan det måste alltid finnas tomma rutor emellan.
- Fartygen måste ligga helt inom spelplanen.

Programmet skall efter varje skott:

- Skriva ut ett felmeddelande om skottet var ogiltigt, dvs utanför planen eller rutan redan beskjuten. Användaren får i detta fall skjuta på nytt.
- Skriva ut resultatet av skottet, träff eller bom.
- Om det blev en träff och fartygets samtliga rutor är träffade är fartyget sänkt. Detta ska meddelas användaren. Dessutom ska programmet markera "beskjuten" på alla kringliggande rutor eftersom det inte kan ligga några fartyg där.

Användaren ska via menyer kunna göra sin olika val. Huvudmenyn kan se ut så här:

```
Dina valmöjligheter (1-3):
1) Beskjuta fiendefartyg
2) Fuska lite, tjuvkika på fiendefartygen
3) Avsluta
Ditt val (1-3):
```

De olika menyerna innebär:

Beskjutning mot fiendefartygen: På skärmen ska spelplanen och skottmarkeringarna visas i en schackbrädesliknande figur. Användaren ska få skjuta ett eller flera skott och därefter återgå till huvudmenyn. Under spelplanen ska träffprocenten skrivas ut.

Fuska lite, tjuvkika på fiendefartygen: Alla spelare har olika moral och alla måste tillfredsställas. Spelplanen presenteras med fiendefartygen fullt tydligt markerade tillsammans med skottmarkeringarna.

Avsluta: Användaren ska få möjlighet att titta på de ej sänkta fartygens lägen före avslutning. Spelplanen presenteras på samma sätt som vid tjuvkiken.

Du bestämmer, som programkonstruktör, enväldigt spelplanens storlek (t ex 8x8). Likaså bestämmer du fartygens antal samt storlek, vilket presenteras i början av programmet (det ska finnas fartyg av olika längder, t ex 1 - 5 rutor). Du skall programmera på ett sådant sätt att man i efterhand lätt kan förändra fartygsuppsättningen. I presentationen av spelplanen ska det tydligt framgå:

- Rutnätet, varje koordinat har en egen ruta, se bilden nedan.
- Resultaten av skotten, t.ex.

  Träffar markeras med '#'

  Bommar med 'o'

  Fartyg , (ej träffade) med 'X'

- Koordinater i kanterna.

Tips: För att representera rutornas tillstånd använder vi oss av någon lämplig konvention. T ex kan heltal ge information om rutan:

0 Rutan ej beskjuten, inget fartyg ligger här.

1 Rutan ej beskjuten, del av ett fartyg ligger här.

2 Rutan beskjuten, bom.

3 Rutan beskjuten, träff.

Det kan vara praktiskt att också lagra mer information för varje ruta. Om det i rutan ligger en del av ett fartyg kan man t ex vilja veta hur långt fartyget är och på vilken ledd det ligger (horisontellt eller vertikalt).

I filen `random` får du tips till slumpfunktionen, som du använder till att slumpa fram fartygens positioner. För att på ett enkelt sätt kunna presentera och ändra fartygens storlek kan du skriva ihop en textfil med fartygens längder. Den kan se ut så här:

```
% Format: heltal som anger fartygens längd (antal rutor)
5 3 2 1 1
```

Presentationen ska vara tydlig.

Extrauppgift, betyg A: Gör ett grafiskt gränssnitt där man klickar för att beskjuta.

Extrauppgift, betyg B: Programmet ska uppdatera en high-score mellan körningarna. Om spelaren är bland de tio bästa (träffprocenten är måttet på spelarens skicklighet) ska namn efterfrågas. Listan lagras efter körningen på en textfil. Denna läses in vid ny programstart. Listan ska sorteras.

Datafiler och hjälpfiler: `random`