

Nästan ett MUD – mer av ett SUD

MUD står för "Multi User Dungeon" och är en benämning på textorienterade rollspel där man spelar en person som utforskar en värld och interagerar med andra personer i världen. Som du kan se på t.ex. <http://www.topmudsites.com/> finns det hundratals "muddar" med olika världar, syfte och stämning.

Om du inte är bekant med MUD-begreppet gör du klokt i att följa länken ovan eller på annat sätt bekanta dig med det. Det kommer förmodligen att hjälpa dig att förstå resten av uppgiftstexten.

PollaxMUD

MUD-värden som du skall bygga skall vara en kopia av Polacksbacken. För enkelhets skull begränsar vi oss till plan 2. Rummen i datormodellen är sammankopplade och det går att gå mellan dem på samma sätt som i verkligheten. Korridorer modelleras som rum.

Avataren

Avataren i spelet är en ung modig student av obestämt kön som traskar runt i MUD-världen. Avataren har följande egenskaper:

- Oavslutade kurser. En lista av oavslutade kurser som avataren är registrerad på. Den är tom i starten.
- Avslutade kurser. En lista av avslutade kurser som avataren är registrerad på. Den innehåller 6 kurser från början.
- HP. En siffra mellan 0–180 som startar på 60.

Avataren har därutöver en ryggsäck som rymmer 10 liter.

Uppdraget

Avatarens uppdrag är att ta ut sin examen. Examensbeviset vaktas av Sfinxen som gömmer sig i något rum. För att få ut examensbeviset måste avataren leta upp Sfinxen, samla 180 HP och ha en tom lista av oavslutade kurser. Går man in till Sfinxen med 180 HP eller mer har man klarat spelet.

HP samlar man genom interaktion med varelser i andra rum.

Rum

Man kan gå runt i rummen med enkla terminalkommandon. När man går in i ett rum får man en textuell beskrivning av rummets utseende, vilka vägar ut ur rummet som finns, eventuella föremål i rummet som man kan ta eller interagera med, samt eventuella varelser i rummet.

Kommandot för att gå ut ur ett rum är "go" och sedan ett väderstreck. Vi gör en förenkling och utgår från att inget rum har fler än en dörr i samma väderstreck.

Vissa dörrar är låsta och kan bara öppnas med hjälp av nycklar (se nedan).

Föremål

Det finns ett antal föremål spridda i rummen i spelet.

1. Nycklar. Alla nycklar passar i alla dörrar, men en nyckel kan bara användas för att låsa upp en dörr – sedan är den förbrukad. För att använda en nyckel skriver man "use key with" väderstreck.
2. Böcker. Böcker hjälper avataren att klara av kurser. En bok är knuten till en särskild kurs, se interaktion med varelser nedan.

För att plocka upp ett föremål skriver man "pick up" och sedan namnet på föremålet. Föremål tar olika mycket plats. En nyckel tar 1 liter och böcker 1–3. Om ryggsäcken som rymmer 10 liter blir full kan man ta bort föremål med "drop" och namnet på föremålet. Kommandot "inventory" listar innehållet i ryggsäcken.

Varelser

1. Lärare. Lärare bor i olika rum. En lärare är knuten till en särskild kurs som också är knuten till en bok och till ett antal HP.

Om man går in i ett rum där det finns en lärare som är knuten till en kurs som är i studentens lista över oavklarade kurser är det 75% chans att läraren ställer en fråga på kursen. Denna fråga är en flervalsfråga (3 alternativ). Om man svarar rätt får man så många poäng som kursen avser; om man svarar fel händer ingenting.

Om läraren istället skulle vara knuten till en avklarad kurs är det 50% chans att denne ställer en flervalsfråga (3 alternativ). Svarar man fel läggs kursen tillbaka i listan över oavslutade kurser och kursens HP försvinner från studentens HP.

Om läraren är knuten till en kurs som inte finns i någon av studentens listor kan man välja att registrera sig på kursen. Detta görs med "enroll" och kursens namn.

Vid flervalsfrågor har man en fördel om man har kursboken. Då försvinner ett av de felaktiga alternativen.

2. Studenter finns litet här och var i de olika rummen. Varje student går en kurs och har klarat en annan. Om man kan ge studenten kursboken för den kurs som studenten går kan man få något i utbyte: antingen kursboken för den andra kursen, eller rätt svar på lärarens fråga för den kursen.

För att prata med en student används "talk" och studentens namn. För att byta böcker med en student används "trade" och namnet.

Sfinxen

Är stationär under hela spelet i något rum med bara en ingång. Sfinxen sitter på examensbeviset som inte kan ryckas loss.

Sfinxen kan man bara interagera med på två sätt: "talk" och "graduate". I det första fallet kommer Sfinxen att dela med sig av någon irrelevant visdom; i det andra fallet kommer Sfinxen att syna avataren för att se om hen uppfyller alla kriterier för att få ut sin examen. Gör hen det så är spelet slut och en fint diplom skrivs ut över avklarade kurser. Om inte händer ingenting.

Skapandet av världen

Världen skall finnas beskriven i en radorienterad textfil där varje post är ;-separerad. Formatet är:

1. Rummets namn
2. Namnet på rummet vartåt rummets norra öppning vetter eller X om ingen finns
3. "- östra "-
4. "- södra "-
5. "- västra "-
6. True om rummets norra öppning är olåst, annars false
7. "- östra "-
8. "- södra "-
9. "- västra "-

Nedanstående deklarerar alltså två rum, "Room 1357" och "Hallway 4" som är sammankopplade i väster respektive öster med en låst dörr.

```
Room 1357; X; X; X; Hallway 4; X; X; X; False
Hallway 4; X; Room 1357; X; X; X; False; X; X
```

Vilka kurser som finns, deras kursböcker och HP finns beskrivet i en annan radorienterad, ;-separerad textfil, t.ex.:

```
Folklivsforskning 101;Kaffekokning i pumpa av ohärdat glas;10
```

Vilka böcker som finns och vilka som skrev dem finns beskrivet i en radorienterad, ;-separerad textfil, t.ex.:

```
Kaffekokning i pumpa av ohärdat glas;Annie Melin;1965;2
```

Den sista siffran avser hur många "liter" boken tar och begränsar hur många böcker man kan bära omkring på samtidigt.

Ett lämpligt antal lärare och studenter bör slumpas fram och placeras mer eller mindre slumpvist på kartan. För varje låst dörr på kartan bör 1,5 nycklar placeras ut, också de slumpvist. (Vi gör en förenkling och hoppas att detta inte leder till ett olösligt spel.)

Objektorienterad modellering

Modellera alla aspekter "naturligt". Det betyder bland annat att varje rum på Polacksbacken skall representeras som ett separat objekt med föremål som kan läggas till eller tas bort, studenter och lärare som skulle kunna komma och gå, etc. Man kan modellera att två rum sitter samman med ytterligare ett objekt, vilket är en naturlig plats för placering om information om öppningen är öppen eller stängd.

Modellera alla varelser som en klasshierarki med Varelse som superklass. Fundera på om avataren platsar i denna hierarki.

Med en vettig design är det enkelt att komma på nya features eller göra utökningarna. Har man en bra modell blir det enkelt att implementera denna uppgift som en faktiskt MUD.

Notera

Denna specifikation är avsiktligt lös och lämnar många designbeslut över till er. Minns att det viktiga är inte vad programmet ni skriver gör, utan hur ni skriver programmet och dess objektorienterade design.

Frivilliga utökningar

1. Implementera spelet som en faktisk MUD, d.v.s. det finns en central server med flera klienter uppkopplade mot sig. Flera avatarer kan då gå runt på Polacksbacken och samarbeta. Om två avatarer är i samma rum bör de kunna tala med varandra på textchatt. De kan också stjäla böcker av varandra med kommandot "steal" som stjälar en slumpvis bok. Enda sättet att märka att man blivit bestulen är med hjälp av "inventory"-kommandot.
2. Simulering. Utrusta avatar-klassen med olika beteenden (se t.ex. strategy pattern för inspiration). Låt sedan spelet ha en puls som går och vid varje puls gör alla avatarer en handling. Vad som händer skall skrivas ut på skärmen.

Man kan tävla genom att skriva olika beteenden och låta dem race:a mot varandra i mudden.
3. Använd XML eller YAML för att spara data istället för det betydligt sämre sättet som specificerats här.