

Bots!

Bots! är ett spel för två eller fler spelare. Spelets syfte är att lära ut grunderna i procedurell programmering på ett kul och underhållande sätt.

Varje spelare har en robot, som ställs ner på en spelplan bestående av hexagoner (sexkantiga ”rutor” – för enkelhetens skull kallar vi dem för ”rutor”, även om vi då menar hexagoner). Roboten ska sedan programmeras till att leta upp och förstöra andra robotar.

För att spela Bots! behövs papper och penna, en spelplan, två pjäser per spelare och en sexsidig tärning.

Roboten

Får vi presentera stridsrobot m/01 ”Gladiator”, det senaste inom underhållningsvåld!

Robot m/01 är en larvbandsdriven robot med en avancerad mikroprocessor, en tjugosex megawatts plasmalaserkanon, tio milliflux pansar och fler sensorer än man egentligen behöver ha koll på. Roboten kan utföra ett antal olika kommandon, närmare bestämt vrida sig, köra framåt, köra bakåt, spana med sensorer, skjuta och lagra variabler. Mikroprocessorn klarar av iterationer (upprepningar) och selektioner (urval). Sensorer gör att mikroprocessorn kan rikta vapnet, och den kan även känna igen allierade robotar. Således kan även lagtävlingar utkämpas.

Robot m/01 är standard inom Tävlingsfederationen för robotdueller, och alla robotar är likadana. Eftersom Tävlingsfederationen avser att belöna intelligens och skicklighet snarare än eldkraft och pansar måste deltagarna programmera sin robot så intelligent som möjligt.

Grunden

Reglerna i Bots! är mycket enkla: följ programmet. När alla fienderobotar är förstörda koras en vinnare, nämligen den robot eller det lag som står kvar på banan.

Turordning

En spelare börjar spelet och gör sitt drag. Efter honom är det spelaren till vänster som gör sitt drag, och så går turordningen runt varv efter varv tills matchen är avgjord eller spelarna tröttnar.

Programsatser

Grunden i Bots! är att kombinera programsatser på ett intelligent sätt för att jaga och förstöra motspelarnas robotar. En programsats är en instruktion i robotens mikroprocessor, och representeras av en liten ruta. Det finns tre sorter: sekvenssatser, iterationssatser (upprepningar) och selektionssatser (urval). Dessa arrangeras på olika sätt för att skapa ett program.

Programpekare

Man flyttar oftare i programmet än på spelplanen. För att veta var i programmet man står använder man en programpekare. Pjäsen på spelplanen visar var roboten är, medan den andra pjäsen är programpekaren, som står på den programsats som ska utföras.

Programregistret

Programregistret håller reda på lite data om roboten, till exempel vilka robotar som är utpekade, om den ser andra robotar, eller om roboten har krockat.

Flytta roboten

Roboten flyttas *om och endast om* en programsats säger att man får flytta roboten. Robotar kan flyttas framåt eller bakåt, eller vridas vänster eller höger, beroende på programsatser. Man flyttar alltid en ruta framåt eller bakåt, och man vrider roboten ett ”hack” åt vänster eller höger. Roboten ska alltid peka mot en av rutans kanter.

Skjuta

Roboten kan göra två saker med kanonen: peka ut ett mål (dvs sikta på målet) eller skjuta ett mål. Ett utpekad mål följs av kanonen automatiskt. Om man skjuter utan att ha ett utpekad mål så går skottet rakt fram. Om man har ett utpekad mål så går skottet mot den utpekade roboten.

Det är inte helt säkert att kanonen träffar. Slå en tärning. Om tärningen visar lika många eller fler ögon som antalet rutor mellan den avfyrande roboten och dess mål så är det en träff, och målet är förstört. Om antalet ögon är färre än antalet rutor mellan den avfyrande roboten och målet så är det en miss, och målet klarar sig.

Om man inte har pekat ut ett mål går skottet rakt fram. Finns det en eller flera robotar i någon av rutorna rakt fram så skjuter kanonen på det närmaste målet.

Förstörda robotar

Förstörda robotar är skrot, och tas bort från banan. Det är ingen katastrof om ens robot blir skrot – robot m/01 är billig. Det är bara att skaffa en ny och kräva revansch med ett smartare program.

Vinnare

Vinnare är den robot eller det lag som är sist kvar på banan när alla andra robotar eller lag är förstörda.

Programmet

Programmet kan ritas på papper eller läggas ut med kort som sedan förbinds med linjer (det rekommenderas att man i så fall använder lite häftmassa eller tejp för att fixera korten). I båda fallen ska programmen se ut på samma sätt. Kontrollen av detta ska vara strikt så att inget program är felaktigt. Skulle det vara felaktigt är det ingen panik – det går i så fall att skriva om programmet.

Varje programsats motsvaras av ett kort. Om man inte använder kort, utan ritar programmet för hand, så måste varje ruta i programmet se likadan ut som motsvarande kort. Det finns tre sorters kort: sekvenskort (programsatser som gör något), selektionskort (urval) och iterationskort (upprepningar). De två sista styr programmets flöde, medan den första kategorin faktiskt gör något.

Flytta framåt (sats)

Roboten flyttar en ruta framåt.

- Om rutan är tom flyttar roboten in i den.
- Om det är en vägg i vägen stänger roboten väggen och står sedan kvar. Krock indikeras.
- Om en annan robot finns framför den egna roboten knuffas den andra roboten bort ett steg i den egna robotens förflytningsriktning, varefter den egna roboten flyttas framåt.
- Om den andra roboten ska knuffas in i en ruta där det står en tredje robot kan den inte knuffas, och den första roboten orkar inte knuffa båda två. I så fall står samtliga robotar kvar. Krock framåt indikeras.

Flytta bakåt (sats)

Roboten flyttar en ruta bakåt.

- Om rutan är tom flyttar roboten in i den.
- Om det är en vägg i vägen stångar roboten väggen och står sedan kvar. Krock indikeras.
- Om en annan robot finns bakom den egna roboten knuffas den andra roboten bort ett steg i den egna robotens förflytningsriktning, varefter den egna roboten flyttas framåt.
- Om den andra roboten ska knuffas in i en ruta där det står en tredje robot kan den inte knuffas, och den första roboten orkar inte knuffa båda två. I så fall står samtliga robotar kvar. Krock bakåt indikeras.

Vrida vänster (sats)

Roboten vrids en kant åt vänster.

Vrida höger (sats)

Roboten vrids en kant åt höger.

Peka ut mål (sats)

Det närmaste av ett av robotens tidigare upptäckta mål pekas ut. Indikera i programregistret vilken robot som pekas ut.

- Som närmast betraktas det mål som befinner sig minst antal rutor från roboten.
- Om det finns flera mål att välja mellan så pekas ett slumpvist valt mål ut.
- Om det närmaste utpekade målet redan är utpekad av roboten när roboten pekar ut det, och roboten pekar ut det igen, så väljer roboten nästa mål.

Skjuta (sats)

Roboten avfyrar ett skott mot ett mål i målområdet.

- Om målet är utanför målområdet kan kanonen inte vridas och avfyras mot det målet.
- Om ett mål är utpekad går skottet mot det målet.
- Om ett mål inte är utpekad går skottet rakt fram i robotens körriktning. Finns det ett eller flera mål i den riktningen så går skottet mot det närmaste.
- För att träffa krävs att man slår lika med eller högre än antalet rutor mellan roboten och målet.
- Om avståndet är sju rutor eller mer kan man inte träffa.
- Ett träffat mål är förstört.

Spana framåt efter mål (sats)

Mål som finns i robotens främre sektor upptäcks.

- Som mål betraktas icke förstörda robotar, vare sig de är vänliga eller fientliga, inom tio rutors avstånd.
- Samtliga mål indikeras i programregistret.

Spana bakåt efter mål (sats)

Mål som finns i robotens främre sektor upptäcks.

- Som mål betraktas icke förstörda robotar, vare sig de är vänliga eller fientliga, inom tio rutors avstånd.
- Samtliga mål indikeras i programregistret.

Begär igenkänningssignal (sats)

Roboten begär igenkänningssignal från utpekad mål.

- Om utpekad mål är vänligt (dvs i samma lag som roboten) indikeras målet som vänligt i programregistret.
- Om utpekad mål är fientligt (dvs inte med i samma lag som roboten) indikeras målet som fientligt i programregistret.

Vänta (sats)

Roboten gör ingenting.

Val (sats)

Roboten utvärderar ett antal alternativ, och väljer sedan vilken väg programpekaren ska ta.

Alternativ (selektion)

Roboten väljer en sekvens under alternativet. Varje alternativ hör ihop med ett enda val. Sekvensen som hör till alternativet placeras till höger om alternativet, så att alternativet blir den första satsen i en ny sekvens.

- Om ett villkor hör till alternativet läggs det ovanför alternativet.
- Det sista alternativet, och endast detta, kan sakna villkor.
- Villkor utvärderas från vänster till höger. Om ett villkor stämmer väljs detta alternativ. Programpekaren flyttas omedelbart till första satsen i sekvensen vid det första villkor som stämmer.
- Om inget villkor stämmer och det sista alternativet saknar villkor väljs det sista alternativet. Programpekaren flyttas omedelbart till första satsen i sekvensen under alternativet.
- När sekvensen vid alternativet är slut flyttas programpekaren till satsen efter valet som alternativet hör ihop med.

Upprepa tills (iteration)

En sekvens under iterationen upprepas tills ett villkor uppfylls. Sekvensen placeras under iterationen med första satsen rakt under iterationen.

- Varje iteration har ett villkor. Villkoret placeras ovanför iterationen.
- Om programpekaren hamnar på iterationen utvärderas villkoret.
- Om villkoret uppfylls flyttas programpekaren omedelbart till satsen efter iterationen.
- Om villkoret inte uppfylls flyttas programpekaren omedelbart till första satsen i sekvensen under iterationen.
- När sekvensen är slut flyttas programpekaren till iterationssatsen.

Upprepa så länge som (iteration)

En sekvens under iterationen upprepas så länge som ett villkor uppfylls. Sekvensen placeras under iterationen med första satsen rakt under iterationen.

- Varje iteration har ett villkor. Villkoret placeras ovanför iterationen.
- Om programpekaren hamnar på iterationen utvärderas villkoret.
- Om villkoret inte uppfylls flyttas programpekaren omedelbart till satsen efter iterationen.
- Om villkoret uppfylls flyttas programpekaren omedelbart till första satsen i sekvensen under iterationen.
- När sekvensen är slut flyttas programpekaren till iterationssatsen.

Villkor

Villkor är vad som styr programmet. Följande villkor finns att välja mellan.

Utpekat mål är vänligt

Utpekat mål finns åt vänster

Utpekat mål finns åt höger

Utpekat mål är fientligt

Krock är indikerat

Inget mål är utpekat

Utpekat mål är inom X rutors avstånd

Utpekat mål är bortom X rutors avstånd

Programmering

Alla spelare börjar med att programmera sina robotar. Programmeringen sker genom att man skapar sekvenser av satser. Dessa satser kan antingen läggas ut i form av kort, eller ritas i form av rutor på ett papper. I det senare fallet måste det finnas plats för programpekaren att placeras på en sats.

Sekvenser

En sekvens är en serie av programsatser som läggs ut från vänster till höger, ungefär som ett tåg. Sekvenser ska *enbart* ligga i en rad från vänster till höger. Att på något sätt lägga sekvenser uppifrån och ner är fel!

Så, följande krav sätts på en sekvens:

- 1) Det ska vara en serie med programsatser som utförs i ordning.
- 2) Programsatserna ska ligga från vänster till höger
- 3) Programsatserna ska ligga på en rad.

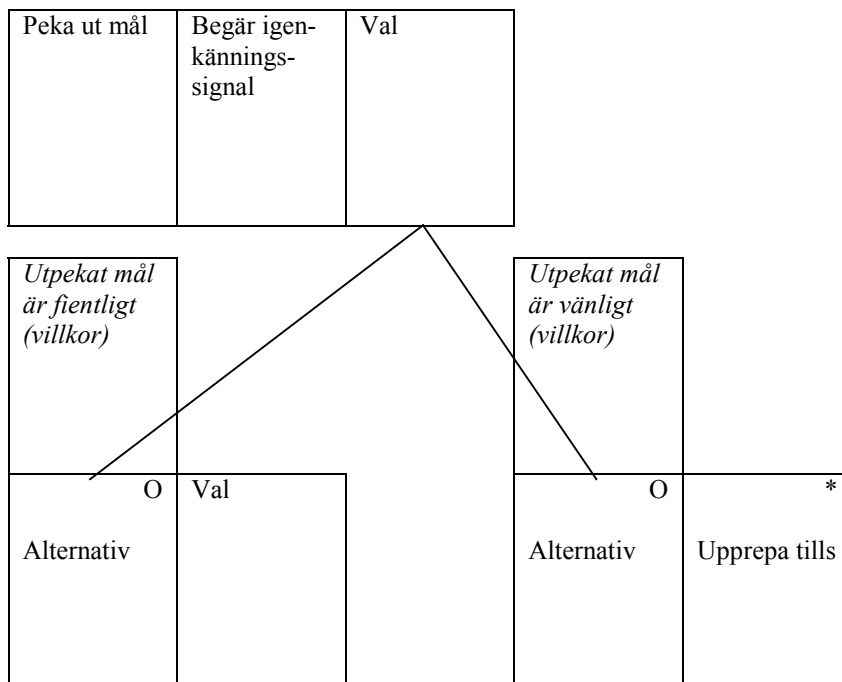
Flytta framåt	Spana framåt efter mål	Peka ut mål	Begär igenkännings-signal	Val

Selektion

En selektion består av minst tre kort:

- 1) ett Val som placeras i en sekvens där selektionen ska utföras.
- 2) ett Alternativ som inleder en ny sekvens, samt
- 3) ett Villkor som säger huruvida man ska gå till alternativet eller inte.

Villkoret skall alltid placeras rakt ovanför Alternativet. Alternativet placeras någonstans i närheten och under Valet, och förbinds med ett streck så att det syns vilket alternativ som hör ihop med vilket val.



Iteration

En iteration består av tre saker:

- 1) Ett Iterationskort i en sekvens, som talar om var i en sekvens en iteration tar plats.
- 2) En sekvens som itereras. Ett streck förbinder iterationskortet med sekvensen som ska upprepas.
- 3) Ett villkor som säger hur länge iterationen ska pågå.

O	*	Skjuta	Vrid höger
Alternativ	Upprepa tills		

Utpekat mål är fientligt (villkor)				
Flytta framåt	Val	Spana framåt efter mål	Peka ut mål	Begär igenkännings-signal

Startprogram

Spelet får inte börja förrän alla spelare har programmerat sina robotar. Det föreslås att man sätter en tidsgräns, så att man inte får programmera hur länge som helst. En kvart till en halvtimme är ganska lagom tid för att programmera.

Omprogrammering

En spelare kan, när det är hans tur, börja programmera om sin robot. Han säger i så fall att han programmerar om roboten, och det blir omedelbart nästa spelares tur. När det återigen blir spelarens tur får han frågan om han är klar. Om han är klar börjar programmet om på första rutan i sekvensen. Om han inte är klar får han fortsätta att programmera om roboten och det blir återigen nästa spelares tur.

Roboten står hela tiden kvar på spelplanen när den programmeras om, och utgör en snäll och fin måltavla.

Flytta framåt	Flytta bakåt	Vrid vänster	Vrid höger	Peka ut mål	Skjuta
Spana framåt efter mål	Spana bakåt efter mål	Begär igenkännings-signal	Vänta	Val	O Alternativ
*	*	<i>Utpekat mål är vänligt (villkor)</i>	<i>Utpekat mål är fientligt (villkor)</i>	<i>Utpekat mål finns åt höger (villkor)</i>	<i>Utpekat mål finns åt vänster (villkor)</i>
<i>Utpekat mål finns rakt fram (villkor)</i>	<i>Krock är indikerat (villkor)</i>	<i>Inget mål är utpekat (villkor)</i>	<i>Utpekat mål är inom 3 rutors avstånd (villkor)</i>	<i>Utpekat mål är bortom 3 rutors avstånd men inom skotthåll (villkor)</i>	<i>Utpekat mål är bortom skotthåll (villkor)</i>
Flytta framåt	Flytta bakåt	Vrid vänster	Vrid höger	Peka ut mål	Skjuta
Spana framåt efter mål	Spana bakåt efter mål	Begär igenkännings-signal	Vänta	Val	O Alternativ
*	*	<i>Utpekat mål är vänligt (villkor)</i>	<i>Utpekat mål är fientligt (villkor)</i>	<i>Utpekat mål finns åt höger (villkor)</i>	<i>Utpekat mål finns åt vänster (villkor)</i>
<i>Utpekat mål finns rakt fram (villkor)</i>	<i>Krock är indikerat (villkor)</i>	<i>Inget mål är utpekat (villkor)</i>	<i>Utpekat mål är inom 3 rutors avstånd (villkor)</i>	<i>Utpekat mål är bortom 3 rutors avstånd men inom skotthåll (villkor)</i>	<i>Utpekat mål är bortom skotthåll (villkor)</i>

Satser och villkor

<div> <div>*</div> <div>Upprepa tills</div> </div>	<div> <div>Tills roboten eller robotens lag har vunnit</div> </div>
<div> <div>Lägg första program- satsen här</div> </div>	