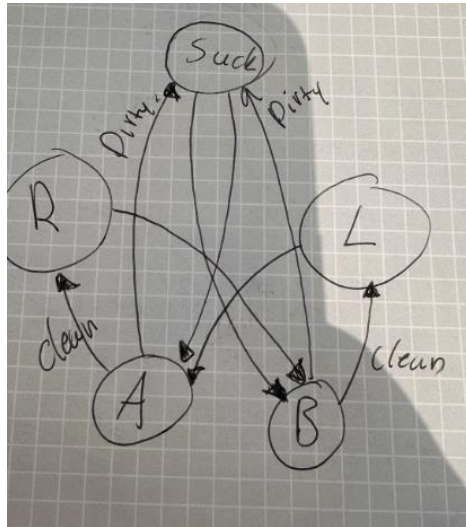


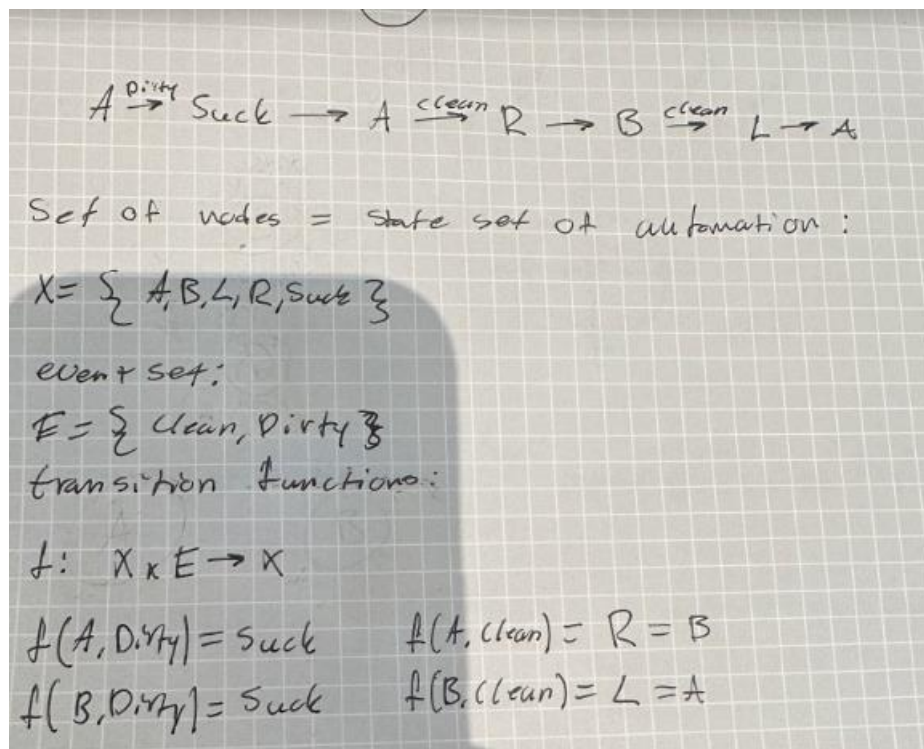
- A.1 What makes an agent autonomous?
 - En autonomy agent är en agent som kan utföra olika tasks självständigt utan human interference, en autonom agent tar alltså egna beslut baserat på dess programmering för att på ett eller annat sätt nå sitt programmerade mål. För att den skall kunna vara autonom krävs även att den självständigt skall kunna skicka vidare denna informationen alltså att den ska ha viss förmåga till kommunikation för att meddela omgivningen om vad agenten kommit fram till. Eftersom att autonomitet precis som boken säger är ett spektrum är det svårt att sätta fingret på exakt vad som krävs för att en agent skall vara autonom men dessa två saker tas upp av boken som exempel på autonomitet.
 - Boken säger att ett objekt i ett objektorienterat programspråk kan ses som en "actor" och att dessa kan ha människolik beslutsförmåga. Boken menar på att det kan se ut som att objekten frågar andra objekt om information och kommunicerar samt utifrån olika parametrar tar olika beslut. Objekt är alltså inte passiva behållare för information och tillstånd utan de kan ses som agenter för programkodens aktiviteter. Boken menar på att det finns klara likheter mellan objekt samt agenter men att där finns skillnader som är tillräckligt stora för att dessa skall vara olika snarare än att agenter och objekt inte är detsamma.
 - Definitionen bakom ett objekt är i grund och botten enkapslation, där de har kontroll över sitt egna interna state. Dessa objekt och variabler i språk som Java kan deklarerats private då de enbart kan hanteras inom samma klass. Public anses ofta vara ett dåligt sätt att initialisera variabler och objekt. Detta leder till att objektet har kontroll över sitt state men inte dess beteende. Alltså ett public objekt har ingen kontroll över när det skall exekveras då det styrs av andra kallelser från andra håll vilket inte är autonomt. Agenter brukar generellt sätt inte tvinga andra att utföra uppgifter utan de brukar ses som att de frågar om en annan agent kan utföra något.
 - Boken summerar detta genom att säga att objekt gör saker gratis medan agenter gör saker för att det är dess egen vilja vilket leder till att agenter och objekt inte är samma samt att objekt inte i detta fall riktigt kan ses som agenter trots deras imitation och viss autonoma förmåga.
 - Enligt control looppen skulle jag argumentera för att ett objekt inte är autonomt då det inte har ett mål och bara gör, det har inte heller någon form av feedbackloop och konsekvenstänk som man kan tänka sig att en agent har då den bara gör vad den blir tillsagd när den blir tillsagd.
- A.2 What makes an agent autonomous, intelligent, both?
 - Oren Etzioni sa att en intelligent agent är 99% datorteknik samt 1% AI. Generellt sätt brukar man betrakta intelligens som förmågan att lära sig som diskuterades på diskussionen i form av att man betraktas som dum om man gör samma misstag om och om igen därav logiskt sätt är intelligens förmågan att lära av sina misstag och titta på sin omgivning och anpassa sig då dessa kan ses som motsatser.

- Generellt sätt i agenter letar man efter de 3 följande karaktäristiska förmågorna, reaktivitet, proaktivitet samt socialförmåga. De behöver alltså kunna reagera på sin omgivning, utföra beslut som leder dem närmare deras mål samt kunna socialisera sig med andra agenter och maskiner i sin omgivning. Precis som autonomitet så är intelligens också en skala och det är svårt att dra en gräns och ge en ren definition då begreppen har mycket laddning och därav olika betydelser från person till person.
- Största skillnaden jag kan hitta är att i boken står det inte implicit att en autonom agent nödvändigtvis behöver ha den sociala förmågan som en intelligent agent behöver. Detta tolkar jag som att en intelligent agent väljer självmant när den skall kommunicera med andra medans en autonom agent gör detta då det krävs för miljön den befinner sig i. Jag skulle även vilja påstå att en autonom agent till viss del är intelligent då de uppfyller ovanstående kraven till viss mån därav är de på samma skala men inte på toppen av intelligens skalan och därav betraktas inte som sådana.
- Intelligent agenter behöver alltid vara autonoma medans autonoma agenter inte alltid behöver vara intelligenta. Jag skulle anse att en agent som kräver mänsklig input och uppmärksamhet inte är intelligent samt att en autonom agent inte alltid kommunicerar med sin omgivning vilket direkt gör att den inte uppfyller intelligens kraven för sociala aspekten och därmed inte kan kallas intelligent. En autonom agent behöver heller inte lära sig av sina fel och kan göra samma fel om och om igen medans ett intelligent system behöver lära sig och anpassa sig för att nå sitt mål.
- Jag skulle kalla kontroll loopen icke intelligent. Den är alltså autonom och inte intelligent eftersom där inte finns någon form av kommunikation mellan den och andra kontroll loopar utan det är ett slutet system.
- The blob that solves Mazes är väldigt intressant då den fyller alla kraven i form av social, proaktiv och reaktiv. Den reagerar på sin omgivning, den letar efter mat och sprider ut sig i sin omgivning för att nå målet samt den kan förmedla information från en slime mold till en annan. Så utifrån denna enkla definition är den intelligent. Dessvärre är motivationen bakom maten och frivilligheten till att göra det ifrågasättbar. Den bara gör detta för det är den programmerad till att göra rent genetiskt lite som ett java objekt vilket vi kom fram till inte var detsamma som varken en autonom eller intelligent agent i ovanstående paragrafer.
- Research material
 - Den interna representationen av miljön är den modell av verkligheten som agenten utgår ifrån och hur den ser på verkligheten. Det är alltså utifrån denna den vet var den är i miljön samt vad nästa steg skall vara för att uppnå dess designade mål.
 - Dynamiska representationer är bäst då autonoma agenter ofta påverkar sin miljö samt att om det är statiskt blir tasken som utför alltid densamma och inte adaptiva.

Eftersom det är dynamiskt kan agenten reagera på hur dess actions påverkade omgivningen



-
- Statemachine för en dammsugare, R är ett state för att röra sig höger och L för att röra sig vänster. A och B är de olika rummen.
- Om vi befinner oss i A och rummet är dirty så dammsugas rummet och vi går tillbaka till A, om rummet inte är rent återgår vi till suck annars är det clean och vi rör oss till höger och når B, där upprepar processen tills det är clean och vi rör oss då tillbaka till A.
- $E = \{A, B, R, L, Suck\}$, $a = \{Dirty, Clean\}$, $t: E \times a \rightarrow E$



○

