School of Engineering and Information Sciences

MIDDLESEX UNIVERSITY

EXAMINATION PAPER

2010/2011 Winter/Spring term

MODULE TITLE Advanced Topics in Games Development

MODULE NUMBER CMT 3325

MODULE LEADER'S NAME Chris Huyck

Time allowed: 3 hours

Total number of questions: 4 questions

Instructions to candidates: Answer all questions. Each question carries 25 marks.

Materials provided: The book Programming Game AI by Example (by Buckland) is allowed, along with one A4 page marked as notes on the top. Please submit that page with the exam (it will not be marked).

Equipment permitted: none

Total number of pages: 3

No books, papers or electronic device is to be brought into the examination room other than any specified above.

Candidates are warned that illegible scripts will not be marked

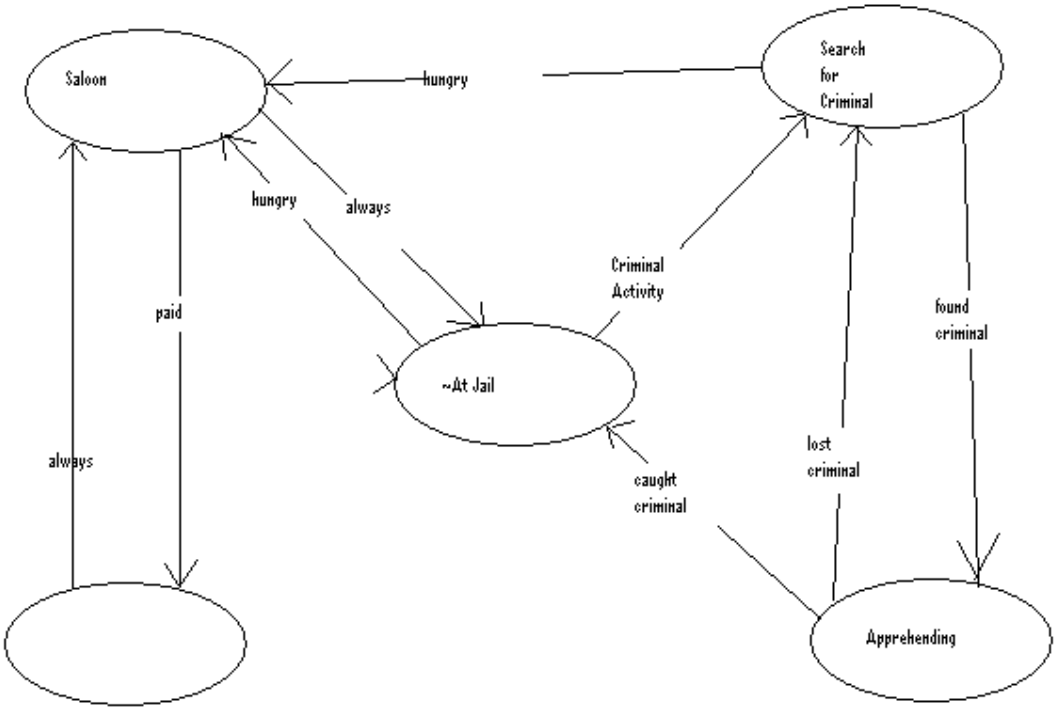
1.AI

(a) In the second lab for this year, we implemented an agent for a Miner, and we did this using a finite state automata. Draw a finite state automata for a Sheriff agent. (Make at least 5 states.)

(10 marks)

Marking scheme:
2 points for initial state
2 points for other states
2 points for arcs
4 points for quality of agent

Sample answer:



(b) The Sheriff lives in a world with 10 cities, and agents can only be in a city, or between two cities. How big is this world's search space?

(6 marks)

Marking scheme:
2 points for cities
2 points for between cities
2 points for right answer

Sample answer:

There are 10 states for the cities. Each city connects to 9 others (not itself) which is 90; however the place between city A and city B is the same as the state between city B and city A, so there are only 45 of those. That makes 55.

(c) Describe an algorithm to visit each state (pseudo code is fine). Explain how efficient the algorithm is.

(9 marks)

Marking scheme:
3 points for an algorithm
3 points for a correct algorithm
3 points for a description of efficiency.

Sample answer:
For each city (move to the next city via the place between the city).
Visit it
For each other city
Visit the place between that city and the other city (returning to the city afterward)

The algorithm is not particularly efficient as it visits edges a bit more than twice. A more efficient algorithm would not visit edges more than twice. Still, it's about the same complexity (the same order) as the most efficient algorithm.

2. Physics

(a) A football is sitting still in the grass and is kicked on its south side. Which way will it move? Paraphrase (or quote) Newton's law that gives you the answer.

(8 marks)

Marking scheme
2 marks for it still being there
6 marks for the paraphrase

Sample answer:
It will move from south to north. This is due to Newton's law that states motion is in the direction of the right line in which that force is impressed is applied.

(b) What's the kinetic energy of a 1 gram bullet travelling at 1000 meters per second?

(9 marks)

Marking scheme
3 marks for E = mvv/2
4 marks for the correct answer
2 marks for correct units

Sample answer:
The kinetic energy of an object is its mass * speed squared / 2.
So E = .001 (kg) * 1000*1000/2 = 500 Joules

(c) Describe a video game where both of these equations might be important. You can use an existing game, or describe your own.

(8 marks)

Marking scheme
2 marks for including part a
2 marks for including part b
4 marks for a reasonable game that includes both.

Sample answer: (obviously, the student's answer will differ significantly from the sample.)

In the real world, if you get hit by a bullet, you get knocked over. In most games, this is ignored as bullets just kill things. However, if body armour is used, the effect will be to knock people over (as the body armour merely distributes the energy). So, a game involving people who are knocked over by bullets could make use of the second equation. If the direction of the bullet is also included, it would also make use of the first portion of the question. Moreover, objects could be moved by bullets, and the game could use different size and speeds of bullets. This could lead to a form of rifle ten pin bowling.

3. Software and AI

(a) A key component of many video games is their agents. Why do agents easily fit into an object oriented paradigm?

(8 marks)

Marking Scheme:
4 marks subclassing
2 data driven
2 communication

Sample Answer:

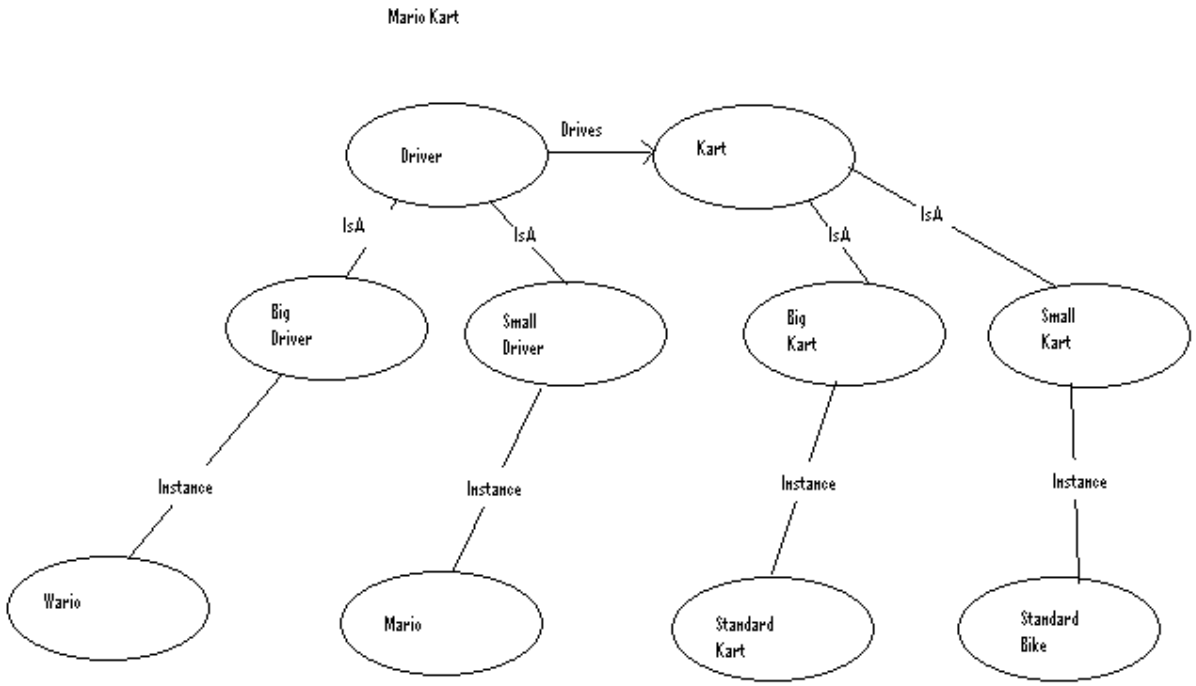
The key to agents is that they can be subclassed. So, a base agent has some general properties. This agent can be specialised into other agents (that have many of the base class's properties) that have new behaviours. These in turn can be specialised again. The underlying data for each instance can also affect behaviour. The overall shell of the program runs independently of the agents; the agents and the environment are affected by communication (actions) from the agents.

(b) Draw a semantic net for agents in a video game (please name the game). Use at least 10 nodes and important relations.

(9 marks)

Marking Scheme:
2 marks isa
3 marks 10 nodes
2 marks other relations
2 marks instance

Sample Answer:



(c) The argument of the superiority of one programming language over another is often said to be irrelevant because they are both Turing complete. Explain what it means for a language to be Turing complete? Why does Turing completeness relate to the superiority of programming languages

(8 marks)

Marking Scheme:
2 marks turing machine (or similar)
2 marks implement any program.
2 marks linguistic equivalence

Sample Answer:

A language is Turing complete if it can implement a Turing machine. If a language is Turing complete, it can implement any program that can be implemented. Since both languages can implement any algorithm, they can both do the same thing.

4. AI

(a) In the lectures, we discussed Artificial Life, with the Sims being a form of artificial life. Describe some basic rules that each Sim might use. Describe rules that particular Sims might use.

(8 marks)

Marking Scheme:
2 marks for rules
3 marks rules for general Sims
3 marks rules for specific Sims
Sample Answer:
Sims need certain things like food and sleep to live; so one rule might be if you are hungry and have food, eat it. Another might be if you are hungry and don't have food get it. A third might be if you are tired sleep. Those are general rules. More specific rules might be user specified. So, a particular Sim might like to play music, and a rule for him might be, if there is spare time, play music. Another rule for the same Sim might be, if I don't have a guitar, and I have money, buy one.

(b) A model might be used to predict population dynamics of for instance hares and foxes. What would be 4 important rules that such a model might use?

(8 marks)

Marking Scheme:
2 marks for each good rule (up to 4)
Sample Answer:
I think the key to this is that foxes eat hares. If there are more hares, the foxes can eat more of them. If the fox eats less hares, he dies, or has fewer children. So rules like each year a fox dies with a 20% chance. If a fox is hungry, his chance of death is 30%. If a fox is mated, and neither is hungry they have 1 cub. A fox catches 10% of the hares in his range. A fox is hungry if he has less than 30 hares a year. A pair of mated hares produces 3 leverets (babies) each year.

(c) What makes simple rules like yours (from question 4a) lead to an interesting game? What are the characteristics of a search space for a fun game that used population dynamics?

(9 marks)

Marking Scheme:
2 points the rules make sense and you can see there effect in the game.
3 marks emergent behaviour
2 marks for a large search space
2 marks for an unpredictable (but not entirely) change in the search space.
Sample Answer:
What makes the game interesting is that there is some correlation between the rules and the behaviour of the game, but the interaction between the agents and the rules leads to unexpected, emergent behaviour. This behaviour may be chaotic, but it is not easily predicted. The size of the search space is large. The evolution of the search space is not predictable but is somewhat regular. So, the user sees the behaviour evolving sensibly, but still can't predict it. Modification of simple variables leads to new and interesting behaviour.