PAPER CODE NO.	EXAMINER : Boris Konev
COMP222	DEPARTMENT : Computer Science Tel. No. 0151 795 4260

Second Semester Examinations 2014/15

Principles of Computer Game Design and Implementation

TIME ALLOWED : Two Hours

INSTRUCTIONS TO CANDIDATES

Answer FOUR questions.

If you attempt to answer more questions than the required number of questions (in any section), the marks awarded for the excess questions answered will be discarded (starting with your lowest mark).

Question 1

1. What is the difference between game engine code and game-specific code? **4 marks**

2. Classify every one of the following

   - rendering,
   - behaviour specific to zombies,
   - message passing,
   - level implementation, and
   - sound playback

   as a part of game engine code or game-specific code. **5 marks**

3. What is a physics engine? Name at least two advantages of using a third-party physics engine and at least two advantages of using an in-house physics routine. **6 marks**

4. Discuss the difference between the traditional Artificial Intelligence discipline and Artificial Intelligence in computer games. **4 marks**

5. A computer game can be defined as a sequence of meaningful choices made by the player in pursuit of a clear and compelling goal. Justify such a definition and give a graphical representation of the classical game structure. **4 marks**

6. The *golden path* in a game in the optimum path for a player to take through the game to experience the game as intended and to get the maximum rewards. Name two methods used by computer game designers to keep a player on the golden path. **2 marks**

Question 2

1. Let $\mathbf{V} = (1, 3, 5)$ and $\mathbf{W} = (2, 4, 6)$ be 3D-vectors. Compute (and show your working)

- **(a)** $\mathbf{V} + \mathbf{W}$ **2 marks**
- **(b)** $2\mathbf{V}$ **2 marks**
- **(c)** $\mathbf{V} - 2\mathbf{V}$ **2 marks**
- **(d)** $\mathbf{V} \cdot \mathbf{W}$ **3 marks**
- **(e)** $|\mathbf{V}|$ **3 marks**
- **(f)** $\mathsf{proj}_{\mathbf{V}}\mathbf{W}$ **3 marks**
- **(g)** $\mathbf{V} \times \mathbf{W}$ **4 marks**

2. Let $\mathbf{M} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, $\mathbf{M}' = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$ and $\mathbf{V} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$. Compute (and show your working)

- **(a)** The product of the two matrices, $\mathbf{MM}'$. **3 marks**
- **(b)** The product of the first matrix and the vector, $\mathbf{MV}$. **3 marks**

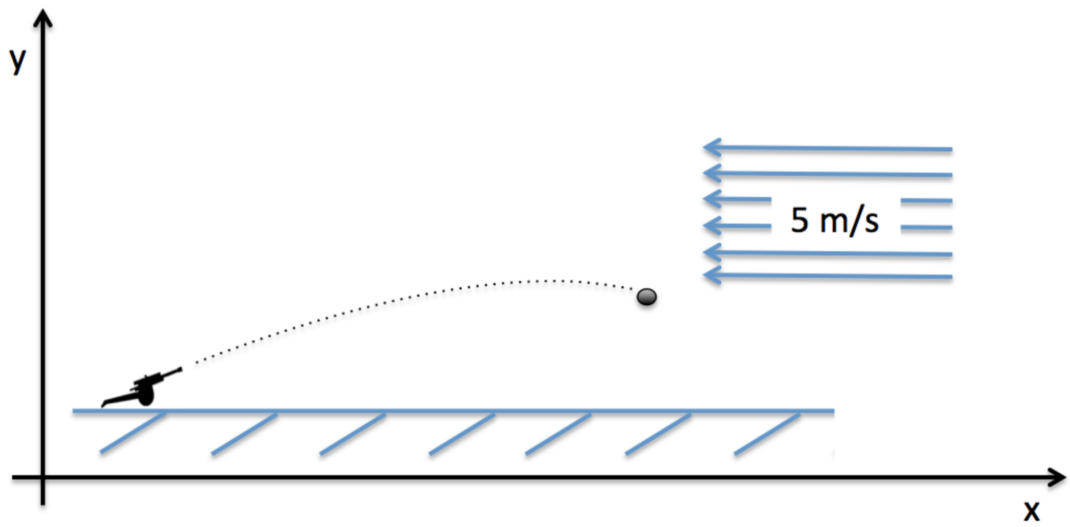
Question 3

1. Modern computer games commonly use scene graphs to represent a graphical scene. Name at least three advantages of this form of data representation as compared with unstructured collections of geometries, light sources, textures, etc. **5 marks**

2. Describe the role of a renderer in a game engine. **4 marks**

3. In this module we studied two major approaches to collision detection: overlap testing and intersection testing. Define these approaches and discuss their advantages and disadvantages. **6 marks**

4. Consider a 2D game, in which a gun fires a cannonball. As part of the gameplay, you are modelling the effect of the air resistance on the cannonball. Additionally, the cannonball moves against the 5m/s wind. The mass of the cannonball is 50kg. The initial speed vector for the cannonball is $(100, 50)$.



Assuming the linear model of drag,

(a) give a graphical representation of all the forces acting on the cannonball as it flies through the air; **2 marks**

(b) describe the discrete motion of the cannonball as a sequence of its positions using Euler steps; **5 marks**

(c) Suppose you are running a computer simulation of the cannonball flight. What method can you use to determine the initial speed vector of the cannonball so that it hits the ground 10 meters from the gun? **3 marks**

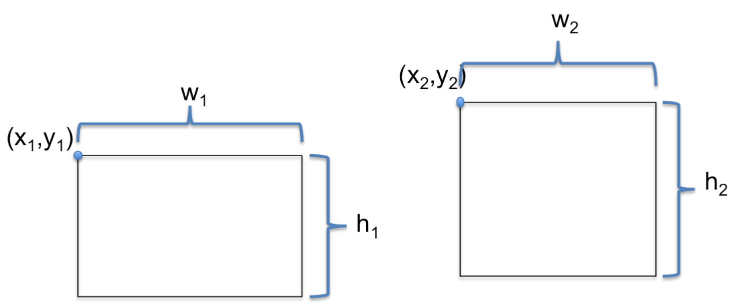
Question 4

1. Poor collision detection can lead to artefacts in computer games. Name at least two undesirable implications of poor collision detection in computer games. **4 marks**

2. Overlap testing in computer games is often approximated with the help of *bounding volumes*: a real shape is being embedded into a simplified geometry, and if two bounding volumes do not overlap, one does not perform an (expensive) triangle-level overlap test.

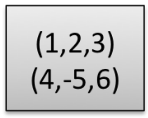
   (a) Simple bounding volume shapes include Axis Aliened Bounding Boxes (AABBs) and Oriented Bounding Boxes (OBBs). What are the advantages of OBBs over AABBs? Are there any significant disadvantages? **4 marks**

   (b) Sketch a method which, given the coordinates of *upper left* corners of two 2-dimensional axis-aligned boxes (x_1, y_1) and (x_2, y_2) and their widths w_1, w_2 and heights h_1, h_2, respectively, determines whether these boxes intersect.



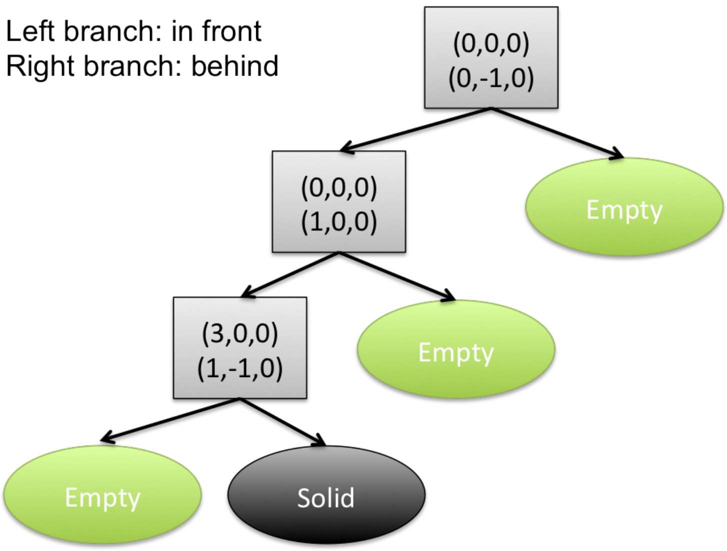
7 marks

3. Recall that a node of a solid-leaf BSP tree can be *solid*, *empty*, or it can be an internal node associated with the plain that partitions the space. In the diagram below, the plain associated with an internal (shown as a box) node is determined by a position vector (first three numbers) and a normal vector (the second line). For example, for the internal node



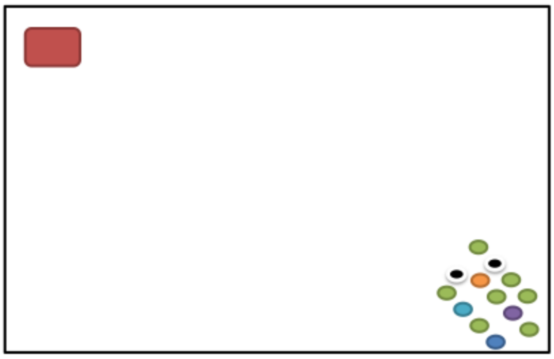
the position vector is (1,2,3) and the plain normal is (4,-5,6).

Sketch the geometrical shape defined by the solid-leaf BSP tree shown below.



Mark clearly on your drawing the position and normal vectors for each plain. **7 marks**

4. In your opinion, what data structure is most suitable to reduce the number of pairwise collision detection tests in a scene where there is one large static object in one corner and a number of small static object in the other as shown below? Explain your reasoning.



3 marks

UNIVERSITY OF LIVERPOOL

Question 5

1. Given the following table representing decisions taken by a human player,

Health	Cover	Ammo	Decision
Healthy	In Cover	With Ammo	Attack
Hurt	In Cover	With Ammo	Attack
Healthy	In Cover	Empty	Defend
Hurt	In Cover	Empty	Defend
Hurt	Exposed	With Ammo	Defend

   and always considering attributes in the order **Ammo**, **Cover**, **Health**, apply the decision tree learning algorithm studied in the lectures to construct the decision tree that, based on attribute values, gives the same decision specified in the table. **6 marks**

2. In computer game AI one can often identify two actors: a virtual player and a game agent.

   - Define what they are and what role they play in computer games.
   - Give an example of both.
   - What is the difference between them?
   - Give examples of when a computer game has a virtual player but no game agents and when a computer game has game agents but no virtual player.
   - How do a virtual player and game agents collaborate?

   6 marks

3. Consider the following behaviour of a fighter game agent. The agent can be in three possible states, *idle*, *patrol*, or *attack*. In the *idle* state the agent remains motionless, in the *patrol* state the agent moves to the next checkpoint, and in the *attack* state the agent attacks another player. If the agent sees the other player, it goes into the *attack* state; otherwise, from being idle it changes, on a timeout, to the *patrol* state and, once completed the move to the next checkpoint, returns to the *idle* state. If the enemy unit is destroyed, the agent goes from the *attack* state to the *idle* state.

   (a) What AI technique is best suitable to represent the behaviour of such an agent? **2 marks**

   (b) Give a graphical representation of this model of agent behaviour. Indicate clearly conditions under which one state changes into another. **5 marks**

   (c) Assume now that you want the agent to show more complicated behaviour: in the *patrol* state the agent patrols four stations $S_1, \ldots, S_4$ in the order $S_1 \to S_2 \to S_3 \to S_4 \to S_1 \to \ldots$ and in the *attack* state the agent goes through three consecutive stages: *approach*, *aim*, *fire*.

   In your opinion, what is the best way to accommodate these modifications to the agent behaviour? Give a graphical representation of the new model of agent behaviour. **6 marks**

Question 6

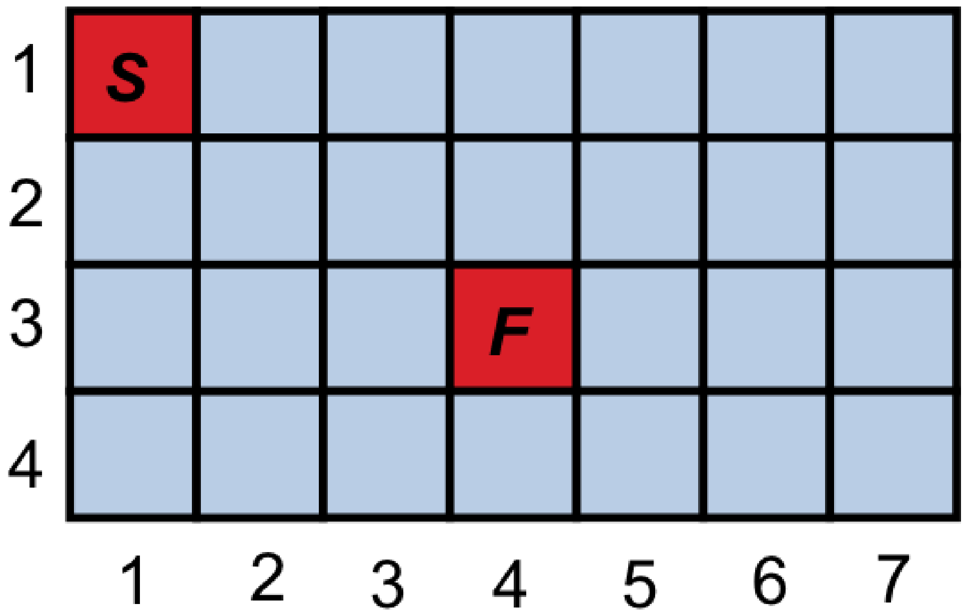
1. Why in computer games is the character motion control routine often considered at two logical levels: steering and pathfinding? Name at least two advantages of such separation. **4 marks**

2. For a turn-based strategy game such as Civilization, and for a first-person shooter game like Quake, which of the following space search structures would you use and why? For the same two games, which one will you not use and why?

   - Regular grids
   - Waypoint Graph Based
   - Navigation Meshes

   3 marks

3. Consider the following tile-based map.



   The only permitted movements are up-down and left-right (if the adjacent tile exists).

   (a) Using Manhattan block distance as heuristic, number the tiles of the map in a way consistent with how the A* algorithm explores the search space to find a path from the start tile (marked with ***S***) to the finish tile (marked with ***F***). **4 marks**

   (b) Suggest a different heuristic to reduce the number of nodes explored by the A* algorithm. Will this heuristic be admissible? **6 marks**

4. Describe the difference between Goal Oriented Behaviour (GOB) and Goal Oriented Action Planning (GOAP) as defined in the lectures. **3 marks**

5. Suppose that a computer character has three goals: Eat(3); Sleep(3); Go_to_bathroom(2). The insistence of every goal is given in the brackets. Which of the following actions should the character choose based on the *overall utility* approach? The effect of every action is given in the brackets.

- Drink-soda (Eat – 1; Go_to_bathroom + 1)
- Visit-Bathroom (Go_to_bathroom – 4)
- Eat-dinner (Eat – 3)
- Take a nap (Sleep – 2)

2 marks

6. In computer games physics simulation often runs at a different speed compared with the drawing updates. Discuss advantages and disadvantages of this approach. **3 marks**