LẬP TRÌNH GAME

GIỚI THIỆU

Thành phần	Thời lượng	Tóm tắt biện pháp đánh giá	Trọng số	Thời điểm
Đồ án		Sinh viên chia nhóm 2-3 người. Chấm điểm quyển báo cáo theo nhóm. Nộp báo cáo và chương trình. Phỏng vấn chấm điểm cá nhân.	30%	**Tuần 15**
Bài tập thực hành		SV làm các bài tập thực hành và nộp lại hàng tuần. GV thực hành tổng hợp và đánh giá. (20%) Seminar Lý thuyết theo topic (10%)	30%	**Tuần 1-14**
Thi cuối học kỳ	120 – 180 phút	Thi trắc nghiệm và tự luận. Được sử dụng tài liệu.	40%	**Theo lịch PĐT**
Tổng			**100%**	

Objectives

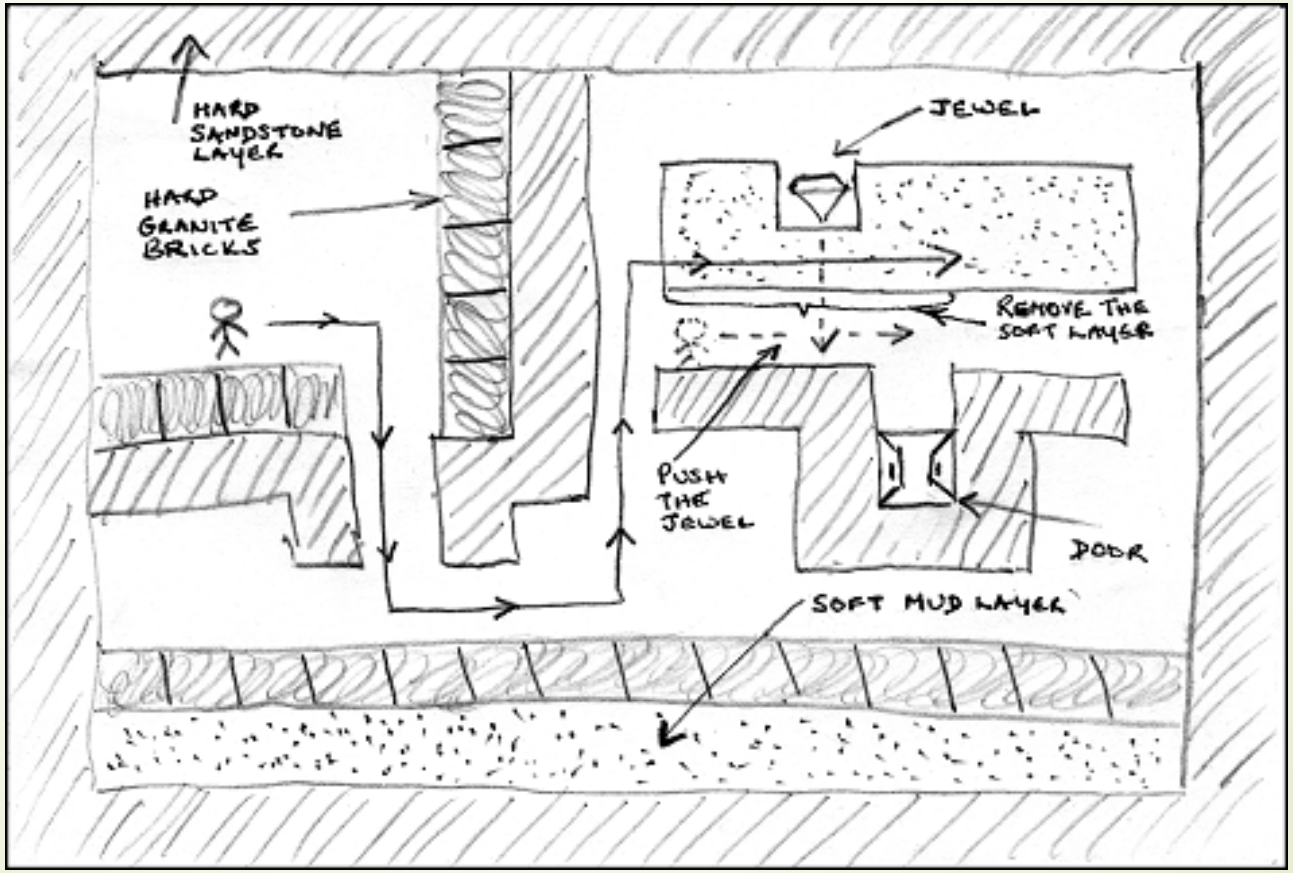
- In this session, you will learn to:
  - Analyze the game requirements
  - Design the game
  - Construct the game

Creating Gameplay Diagram and Bubble Diagrams

- Gameplay Diagram:
  - A gameplay diagram is a rough sketch of the sequence of events that take place at a level.
  - A gameplay diagram shows, a rough sketch of how a level would look like and how game objects are placed in it.
  - Additionally, it will also have flow lines indicating sprite movements.
  - A gameplay diagram have comments describing the user interface and what needs to be done to clear a level.

Creating Gameplay Diagram and Bubble Diagrams (Contd.)

- The following figure shows the gameplay diagram of level 1 for the game Crypt Raider.

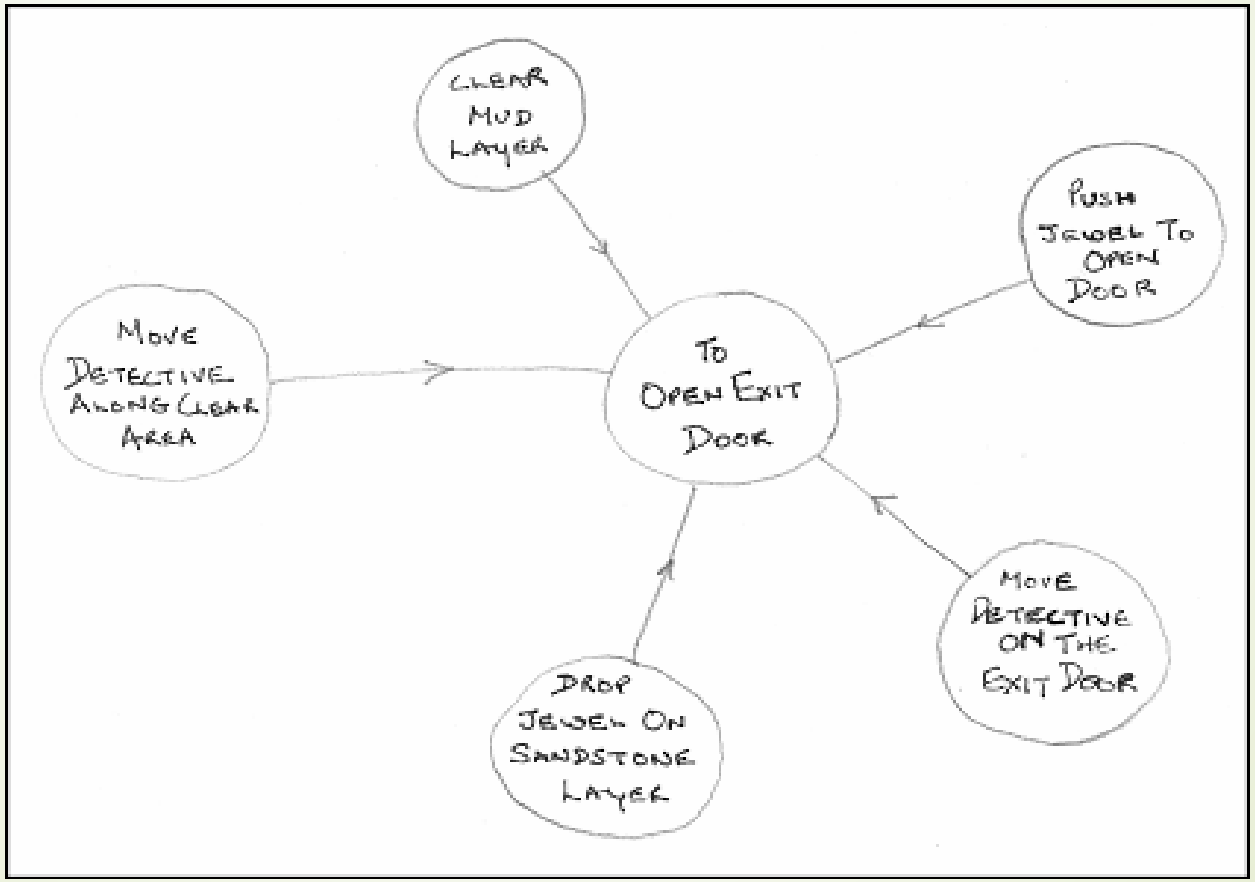


Creating Gameplay Diagram and Bubble Diagrams (Contd.)

- Bubble Diagram:
  - Bubble diagrams consist of a series of circles indicating the areas in the landscape that will be used.
  - The area may map to different parts of the landscape or terrain and may represent a turf area, a shrub border, a hill, or a garden.
  - The circles represent specific locations, and the lines with notes and comments within the circles represent transitions between locations.

Creating Gameplay Diagram and Bubble Diagrams (Contd.)

- The following figure shows bubble diagram.

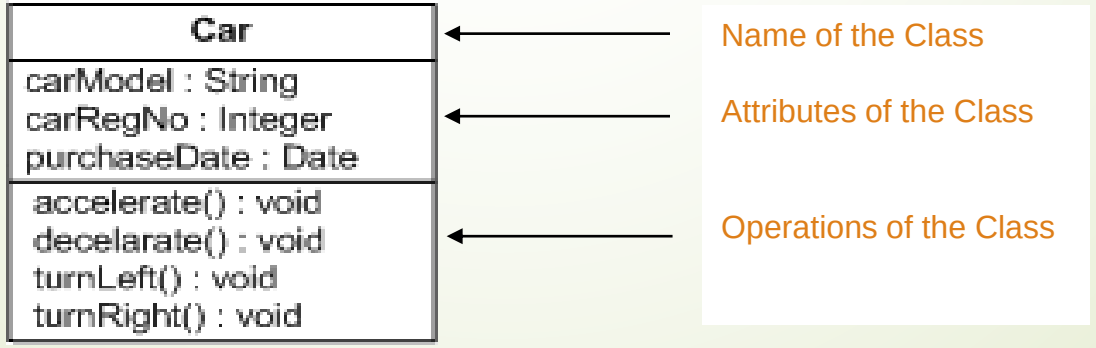


Creating Gameplay Diagram and Bubble Diagrams (Contd.)

- Gameplay Diagrams Vs. Bubble Diagrams:
  - The basic differences between the two diagrams are in the level of detailing.
  - Bubble diagrams offer provision for a very high level of detailing when compared to gameplay diagrams.
  - This makes bubble diagrams more suitable for games that have complex landscapes or terrains.
  - They serve as an excellent tool to map game activities to different parts of the landscape or terrain.

Creating High-Level Class Diagrams for Game Objects

- Introducing Class Diagrams:
  - Unified Modeling Language (UML) provides a class notation to represent classes in class diagrams.
  - The class notation has three compartments:
    - The first compartment depicts the name of the class.
    - The second compartment depicts the attributes of the class.
    - The third compartment depicts the operations of the class.
  - The following figure shows the class notation.



Activity: Designing the Game

- Problem Statement:
  - You have been asked to make a 2-D game prototype of a space shooter game to be called Space Shootout. You are provided with some general design specifications about the game that is as follows:
    - The game starts by displaying a splash screen with the company's logo.
    - This is followed by displaying a menu that has four options: Play Game, Credits, Help, and Quit. Only the first and the last options are selectable. Selection is made by pressing the correct option number on the keyboard.
    - If the play game option is selected then the game screen is displayed that has a spacey background and game statistics. A UFO controlled by the game moves continuously between the left and right boundary of the game window. A Star Fighter is placed at the bottom of the game window.

Activity: Designing the Game (Contd.)

- The Star Fighter can be moved to the left or to the right by a player by pressing the appropriate left or the right arrow key on the keyboard. The Star Fighter is capable of firing missiles at the UFO when the spacebar is pressed.
- The missile disappears as soon as it touches the top boundary of the game window. While the missile is in an active state moving towards the target another missile cannot be fired.
- The objective is to shoot as many UFOs as possible in 3 minutes.
- The game will terminate automatically after 3 minutes displaying the player's scores.

- Based on the preceding specifications, draw up a plan to create the following:
  - Functional Specification Document
  - Technical Specification Document

Construction Phases for a Game

- Pre-Alpha Phase:
  - Pre-alpha build for a computer game is usually a stage when the game is in a prototype stage.
  - It is characterized by many development releases or nightly builds.
- Alpha Phase:
  - Alpha testing, also known as acceptance testing is the pre-release testing of a game internally in a development environment before testing it with outside users.
  - It does not implement the full functionality of the game but satisfies most of the software requirements.
  - The alpha build of the game is the first build delivered to an internal team of software testers.

Construction Phases for a Game (Contd.)

- The software tester completes the alpha testing in two phases.
- In the first phase, white box testing is performed.
- In the second phase, black box testing is applied to the game.

- Beta Phase:
  - The beta phase represents the first version of a computer game that implements all the features as per the requirement documents.
  - This is useful for demonstration to selected customers and stakeholders.
  - The beta phase begins when the developers announce a code freeze and only bugs, software issues and features that have not been implemented will be worked on and fixed at this stage.

Construction Phases for a Game (Contd.)

- Gold Phase:
  - A game is said to be in the gold phase when it is available for commercial release.
  - In this phase, the game is final as a product and all known bugs and issues have been fixed.
  - In the gold state, a game is considered stable and its quality is suitable for sale to end users.

Coding, Testing, and Packaging the Game

- Coding:
  - After the game is designed and the design documents are created, it is time to begin the coding for the game.
- Testing:
  - Unit Testing
  - System Testing
  - Load Testing
  - Portability Testing
- Packaging and Deploying:
  - After the game is successfully tested, it should be packaged using an appropriate installer program, which will make it easily installable on a target computer.

Unity

- Setup
- Demo