

Introduction to Game Programming

Console video game
Off-topic session 1 – Video game insanity

Nacho Iborra

IES San Vicente



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Index of contents

1. Introduction.....	3
2. Project #1: Nim.....	4
2.1. User requirements.....	4
2.2. Technical requirements.....	4
3. Project #2: The big adventure.....	5
3.1. User requirements.....	5
3.2. Technical requirements.....	5
4. Project #3: Pong.....	6
4.1. User requirements.....	6
4.2. Technical requirements.....	6
5. Project #4: Tic Tac Toe.....	7
5.1. User requirements.....	7
5.2. Technical requirements.....	7

1. Introduction

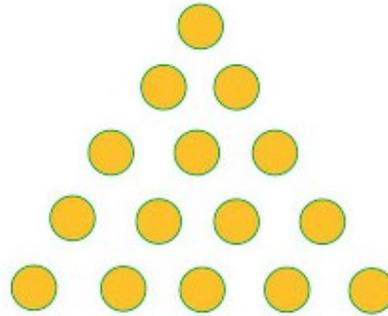
In this off-topic session about console video games, you will have to work in pairs to solve and deliver as many video game requirements as possible, among the video game list shown in this document.

Each video game has a list of user requirements, and it can also have an additional list of technical requirements that you must follow in order to complete the task properly. Besides, every video game has a final price/reward associated to it, so that if you follow the requirements and complete the video game, you will earn this "virtual" money as a reward.

Your final mark in this task will depend on the total reward that you (as a team) can get. Each one of you must work in one of the suggested projects for an hour, and then you must swap your work with your team mate. At the end of the session, you must deliver all your work (i.e. the C# source files) along with a TXT file where you must explain the work done by each team member during each part of the whole session.

2. Project #1: Nim

Reward: €500



Nim is a strategy/intelligence game for two players, in which the game starts with an amount of chips or pieces on a table. Each player can take off from 1 to 3 chips in each turn. The player who takes off the last chip on the table loses the game.

2.1. User requirements

You must implement a single player game, where the computer will act as the second player. At the beginning of the game, you must specify the total number of chips (an integer number between 15 and 50, inclusive). Then, human player always starts the game, and both players (human and computer), will be taking off between 1 and 3 chips in each turn, until the game finishes. Then, you can choose if you want to play again (with the same number of chips than before), or exit the game.

2.2. Technical requirements

Computer player must have some kind of AI in its choices. The game will not have any reward if computer just takes a random number of chips with no reason, or if it always takes the same number of chips with no strategy at all.

3. Project #2: The big adventure

Reward: €500



Conversational video games were very popular in the 80s, along with old computers such as ZX Spectrum or MSX. They consisted in a sequence of scenes with a short text explaining each scene, and a prompt asking the user what to do. If the user chose the correct option, he could move to next scene.

The processing language algorithm could be pretty easy. For instance, if the solution for a given scene was to open a door, you can type either "open door", or "open the door", or "just open the f...ing door"... any sentence which contains the "magical" words ("open" and "door" in this case), in the appropriate order.

3.1. User requirements

You must develop a conversational video game with at least 5 scenes in which the player must face a given test (open a door, kill an enemy, say a secret password...). The game must provide appropriate answers in case the player does not find the correct answer (for instance "I don't know what 'XXXX' means", or "The enemy is getting angry"...

Optionally, you can also provide some kind of "artwork" as background (a sort of image before the text explaining the scene). In this case, your reward might be increased in €100.

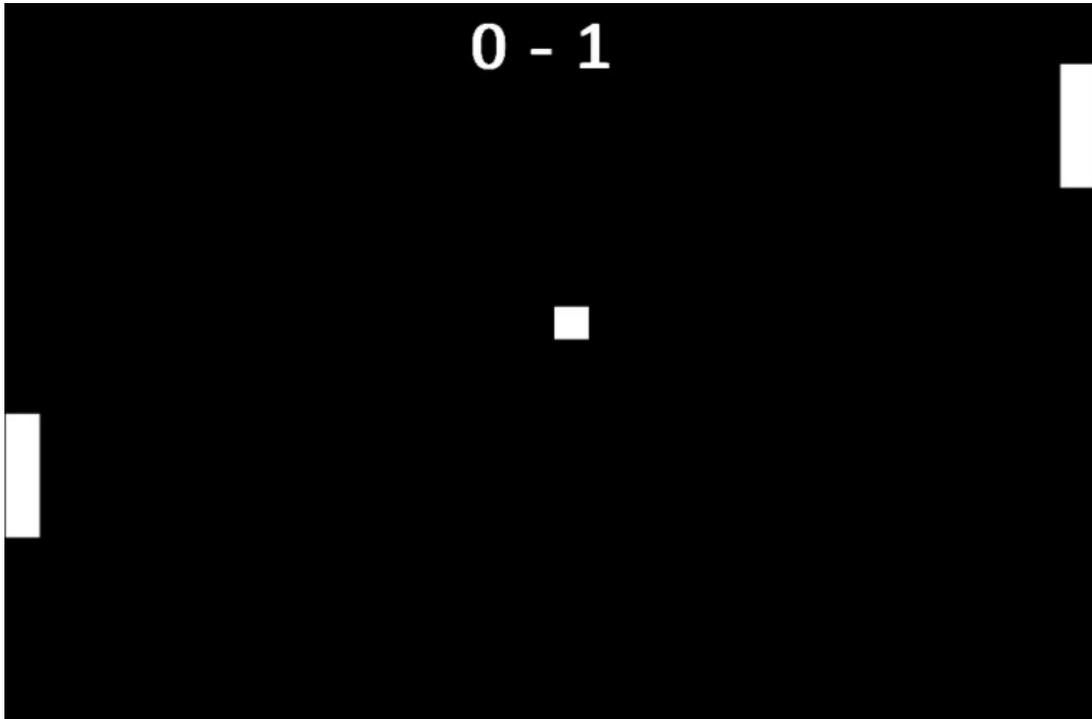
3.2. Technical requirements

You must use string arrays to implement this video game. In these arrays you must store the description of each scene, and you can also store the possible answers to wrong decisions for each scene.

The game must explore all the scenes in order, showing the corresponding description, and moving to next scene (or to the end of the game) if the user finds the way to move on.

4. Project #3: Pong

Reward: €1000



Pong is a very popular game from the 70s made for two players. Each player can move a bar at the left/right of the screen, and they must hit a ball to score a point. This ball bounces against each bar or the screen upper and lower limits. You can see Pong in action [here](#).

4.1. User requirements

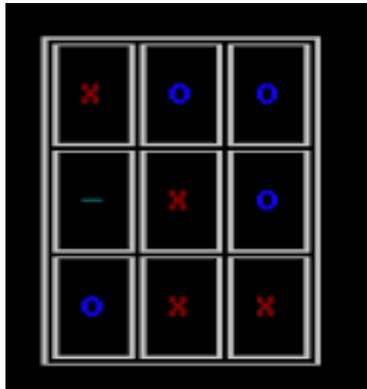
You must implement a two player game in which each player will move a bar with two given keys (up and down). The player on the left (player one) will always start the game, and it will finish as soon as one player gets to 10 points.

4.2. Technical requirements

In order to make the ball move diagonally, you can either make it bounce randomly every time it collides with a user bar, or apply some bounce algorithm, like the one used in Arkanoid.

5. Project #4: Tic Tac Toe

Reward: €700



Tic Tac Toe is also a well known board game for two players. The player who can make a row/column/diagonal with his chips wins the game, and each player can place only one chip in each turn.

5.1. User requirements

You must implement a single player game similar to the one shown in the image above. The human player always starts the game, and it must use the arrow keys to place the cursor in the desired cell. Then, pressing space bar he will drop a chip (blue) in that place, and then the computer must play.

5.2. Technical requirements

You must use an array to store the board game, and you must implement some kind of AI in the computer choices. Remember that, in this game, winning is very important, but not losing is also essential.