# LAB PROJECT – THE HYPERSPACE CHEESE BATTLE

# Do you want a challenge?

**Game of “Hyperspace Cheese Battle”**

Hyperspace Cheese Battle is a game of intergalactic conflict and racing. And cheese. Between 2 and 4 players can play. Each player controls a rocket which they move through space. Moves are controlled by the use of a multi-faceted random value indicating system, otherwise known as dice. Players roll the dice to move their rockets onto space quadrants otherwise known as squares. Certain parts of space are infused with “Cheese Power” which can be used by the advanced technology in the ships to perform special actions.



**Rocket Moves**

Each player takes it in turn to throw one dice. They start off the board at the bottom left hand corner and move their rocket onto the board with their first throw of the dice. At each turn they must then move their rocket that number of squares over the board in the direction of the arrow on that square. If their throw would take them off the board their rocket is not moved.

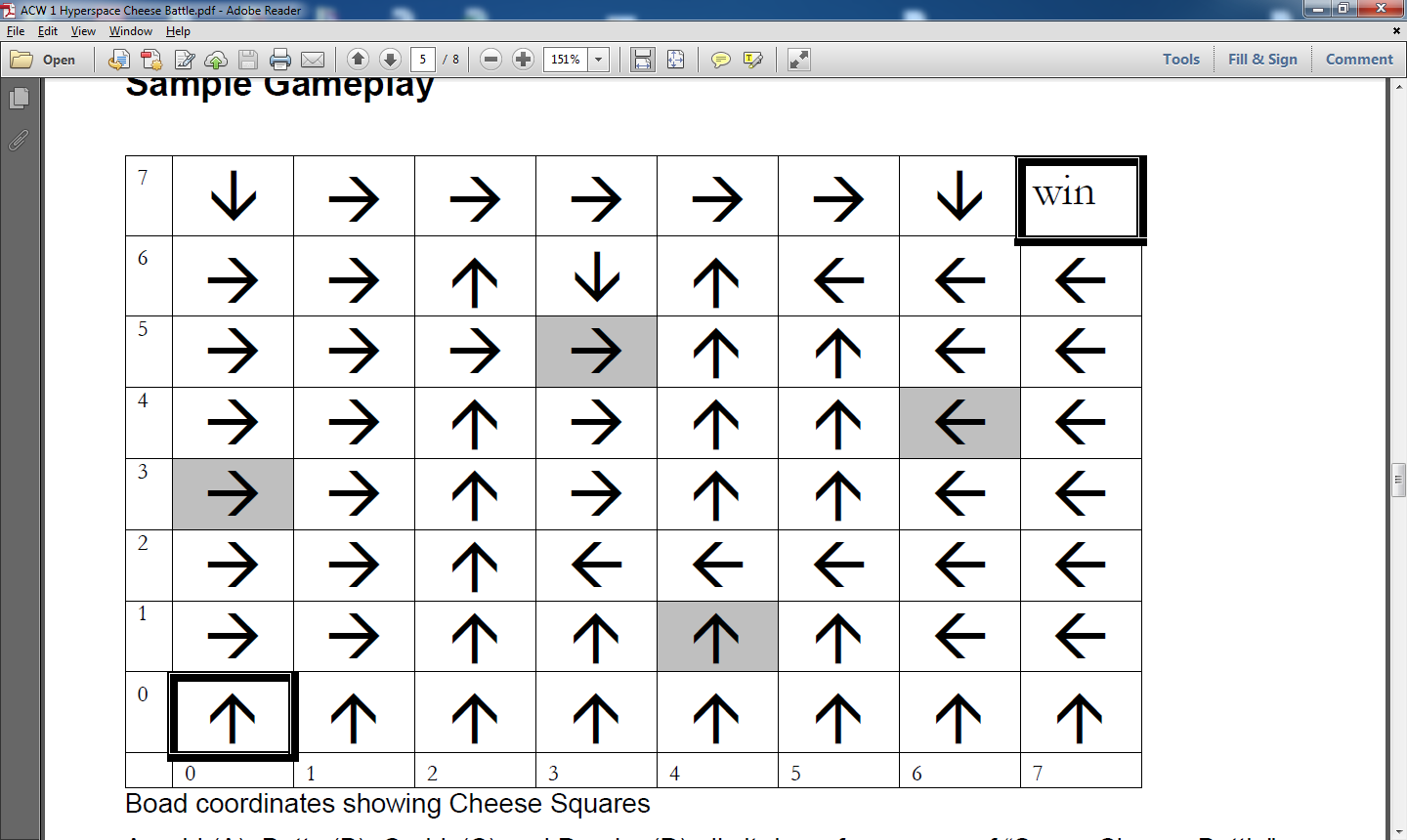
If they land on a cheese square they can perform one of two actions:

* Absorb the cheese power from the square and use it to refuel their engines, allowing them to roll their dice again for an extra move.
* Absorb the “Cheese Power” from the square and create a “Cheese Deathray” which they can fire at any another player. This causes the engines of that rocket to explode, sending it back to the bottom row of the board. The player being sent to the bottom can choose any unoccupied square on the bottom row.

Play continues until a player manages to travel to the square at the top right hand corner. The first player to do this is the winner of the game.

The laws of space and time do not allow a rocket to occupy the same hyperspace as another. If a move would cause a rocket to land on an occupied square the rocket is moved to the next free square in the direction of the arrow on the occupied square. If that square is also occupied the rocket is moved in the direction of the arrow of the next occupied square until an empty square is found.

**Sample Gameplay**



Arnold (A), Betty (B), Cedric (C) and Damian (D) all sit down for a game of “Space Cheese Battle”.

1. A rolls a 2, follows the arrow on (0, 0) and is placed on square (0, 2).
2. B rolls a 2, follows the arrow on (0, 0) and discovers that square (0, 2) is occupied. She follows the arrow on (0, 2) and moves onto square (1, 2)
3. C rolls a 3, follows the arrow on (0, 0) and lands on square (0, 3). This is a Cheese Power square. C decides to roll the dice again and rolls a 4, landing on square (4, 3).
4. D rolls a 3, follows the arrow on (0, 0) and lands on square (0, 3). This is a Cheese Power square. He decides to explode the engines of C’s rocket, and C decides to put their rocket on square (6, 0).

This shows how the game works.

**Game Program**

You are going to write a program which will allow two to four players play a game between themselves. They can move their rockets on their own board but the game will make all the moves for them. The players can continue to take turns until somebody reaches the end of the game. It doesn’t need to actually draw the board; the program will display messages such as:

Jim has thrown a 6 and moves on to square 6, 3

The position of the player on the board should be expressed as a pair of coordinates as shown on the diagram above.

If a player must make a choice the program should ask them to enter their choice of action, for example:

Jim has landed on a Cheese Power Square.

Does Jim want to roll again or explode the engines of another rocket? Enter t (throw) or e (explode):

You will have to design the user interaction so that the players can play the game in the manner described above. The user interface should reject invalid commands.

The game will allow up to four players. The program should ask how many players and accept their names. During the game it should address each player by their name when it is their turn.

**Minimum Specification**

The minimum specification for a program that is regarded as a working “Hyperspace Cheese Battle” game is as follows:

1. Correct acceptance of number of players and their names.
2. The computer should produce a random dice throw for each player and report the result of their move. If the player lands on a “Cheese Power” square the computer must ask the player whether they want to throw the dice again or explode the engines of another player.
3. The program should detect when the winning player has reached the end of the game.
4. The program should allow the user to begin another game when one is completed.

**Suggested Enhancements**

* **Six power**. If a player throws a six they have the option to throw again. If they land on a cheese power square they must use the cheese power rather than have their extra throw. If they throw three sixes in a row their rocket engine explodes they are sent back to the bottom row.
* **Computer players**. The game should have the option of playing with computer players who are either “Angry Allan”, “Speedy Steve” or “Clever Trevor”. Angry Allan will always explode the engines of another player when he lands on a cheese power. He picks the player he wants to swap with at random. “Speedy Steve” will always throw the dice again and “Clever Trevor” will explode the engines of the player furthest ahead, or throw again if he is in the lead.
* **Board display**. The game should display the board and the player positions. The board can be drawn using a character based display or graphically using JXNA (A Java Game Framework similar to C# XNA).