

# 08101 Programming I ACW1 – Hyperspace Cheese Battle

## Tuesday 10<sup>th</sup> December

09:00	Thomas Ablett	Robert Paul Adams
09:15	Ben Adlam	Foysal Ahmed
09:30	Bashar Al-Shamari	Christopher Aldred
09:45	Felipe Renan Christ Alencar	Abdul Rehman Asghar
10:00	Luke Austin	James Bail
10:15	Rhiannon Jade Bailey	James Bakewell
11:00	Kierron Ashley Battle	Alice Rose Beanland
11:15	Connor Berden	Thomas Blears
11:30	Nicholas James Edward Bleasdale	Owen Bones
11:45	Thomas Bottomley	Cameron Bradford
12:00	Sarah Rheanne Bradley	Johannes Brittain
12:15	Alexander Brumwell	James Burgess
14:00	Joe Burnham	Katie Louise Burton
14:15	Paul Butler	Paul Carter
14:30	Fraser Casey	Christopher David Caskey
14:45	Thomas Chestney	Anish Kumar Chopra
16:15	Samuel Cipolat	Glenn Clarke
16:30	Connor Clarkson	Lawrence Clucas
16:45	Conor Colgan	Liam Collins
17:00	James Richard Coyle	Jordan Cranmer

## Wednesday 11<sup>th</sup> December

10:15	Federico Dalla Rizza	Jack Davis	Shaun Charles Lawrence Dervey
11:00	Philip-Mark Devine	Tommy Dockerty	Igor Dojlidko
11:15	Aleksander Wojciech Dul	Albert Dulian	Benjamin Dunn
11:30	Alexander William Dyson	Scott Egerton	Ashley Fairburn
11:45	Joshua Falcon	Jonathan Farrell	Alex Fisher
12:00	James Fisher	Daniel Fleming	Jacob Forbes
12:15	David Fox	James Fox	Christopher Thomas Frankland
14:00	David Andrew Gamble	Joshua Lee Gibson	Jake Aaron Godfrey
14:15	George Goldson	Tom Gothorp	Sean Grady
14:30	David James Graham	Andrew Green	Nicholas Greenham
14:45	Laura Haigh	Andrew Haines	Joshua Hall
15:30	Tomas Gareth Hall	Matthew Hardwick	Andrew Hart
15:45	Matthew Hartley	Adam Kevin Harvey	Daniel Hobson
16:00	Kojo Homiah	Thomas Eammon Hosty- Cundell	Natasha Hotham
16:15	Jack Meriden Hoyle	Wenqing Huang	Anthony Hughes
16:30	William Alan Hutchinson	Nour-El-Din Ibrahim	David Iles
16:45	Martin Iliev	Yuk Long Ip	Viktorija Ivanova
17:00	Richard Jackson	Thomas Harley Jackson	Luke Jarvis

## Thursday 12<sup>th</sup> December

09:00	Matthew Jemmett	Paul Johnson	Joseph Jones
09:15	Adam Christopher Kadow	Mabel Kapangyarihan	Conor Jacob Kaye
09:30	Peter Kemp	Filip Kiedrowski	Rory King
09:45	Jacque Klompas	Joseph Knaggs	Tom Knight
10:00	Andrew Lai	Marcus Lamb	Gareth Lancaster
10:15	Rupert Langford	Johnathan Callum Lauder-Ross	Patrick Lawson
12:15	Liam Christopher Lees	Krishan Marco Madan	Gio Man
14:00	Stefan Mansson	Kuba Maruszczyk	Nezar Marzouk
14:15	Daniel Canlin Masterson	Fernando Manuel Silveira Mateus	Morgan Rose Mills
14:30	Oliver Mills	Mohamed Abukar Moalin Moalin	Cezar Valentin Mocanu
14:45	Anthony Moore	Tom Morgan	Matthew Morris
15:30	Rhian Elisabeth Mulligan	Andrew Peter Murphy	Jerrad Murphy
15:45	Sam Murphy	Matthew Newall	Stuart David Norman
16:00	Rory O'Halloran	John Reginald Oakley	Oluwafemi Adedayo Ogunjimi
16:15	Jack Oldroyd	Muyiwa Olu-Ogunleye	Sulaiman Taiwo Ottun
16:30	Matthew Oxendale	Caitlin Padgett	Henry Parker
16:45	Alexander Pedler	Chamath Perera	Sean Andrew Sidney Phillips
17:00	Robert Phin	William Pitt	Jacob Plaster

## Friday 13<sup>th</sup> December

09:00	Richard Plews	Andrei Popa	Richard Prince
09:15	Armin Pudic	Zbigniew Pysczuk	Daniel Ian Radford
09:30	James William Rawsthorne	Elliott Revell	Sam Richmond
09:45	Daniel Ridgway	Samuel George Rilatt	Maxwell Ritch
10:00	Matthew Robertson	Kieran Robinson	William Robson
10:15	Nicholas Ross	Robert Rowe	Toby Joe Sanders
11:00	Christopher Thomas Sewell	Nathan Simm	Simone Smallwood
11:15	Gregory James Smart	Robert Laurence Smith	Daniel Soares
11:30	Camomile Soper	Elliot Sowersby	Samuel Spain
11:45	Hector Standen	Jacek Stefanowski	Luke Stewart
12:00	Ashley Stopard-Baker	Perry Lee Swain	Joe Swindells
12:15	Willianne Dimpna Caga Tan	Jordan Tavaglione	Aaron Taylor
14:15	Jack Taylor	Jack Alan Taylor	Robert Taylor
14:30	Nathaniel To	Jacob Toner	Joseph Trost
14:45	Mark Vardy	Allan Emmanuel Vargas	Adam Walker
15:30	Sam Waudby	Declan Whiting	Joshua Whorley
15:45	John-Ross Williamson	Ryan Wilson	Duncan Woods
16:00	Andrew Woodward	Eleanor Jay Worsley	Junyi Wu
16:15	Mark Zand	Agisilaos Zervakis	

## Coursework Submission

You will submit your coursework as a single Zip archive using eBridge. The archive should contain:

1. The source code and executable for your program. If you have used Visual Studio to create your program you should include all the Visual Studio project files and directories.
2. The user documentation for your program.
3. The test document for your program.

You should only use the Zip compression tool to create your submission; other compression standards are not acceptable.

## Software Demonstration

As part of your assessed coursework you are required to demonstrate your software. A 15 minute slot has been allocated for you on Tuesday 10<sup>th</sup>, Wednesday 11<sup>th</sup>, Thursday 12<sup>th</sup>, or Friday 13<sup>th</sup> December. This demonstration is **mandatory**. If you do not demonstrate your system you will not be allocated a mark for this piece of coursework, which may well result in your failing the course. If you need to swap your session with another student this is permitted, but you must contact Rob Miles ([r.s.miles@hull.ac.uk](mailto:r.s.miles@hull.ac.uk)) to inform him that change has been made.

## Presenting your software

Each of the markers will have an area of the ITMB lab (room 321) allocated to them. Note that the marking is **not** taking place in the Fenner Computing Laboratory.

You should find a machine in the appropriate area. It is important that you arrive early and allow time to log in to the network and open up your program. Make sure that you have a copy with you or on your drive G: as we will not be using eBridge to view the delivered code.

Before the presentation you should ensure:

1. You have a copy of your software running for demonstration.
2. You have an editing window open on the source of your software.
3. You have an editing window open on the documentation of your software.
4. You have an editing window open on your test document.

You will be required to demonstrate your game running and then show the source and documentation. The marker may ask further questions and discuss your program with you. They will fill in a mark sheet as shown in the specification for this Assessed Coursework. The categories on the mark sheet are aligned with the ones in the marking scheme for this module.

If you are using Visual Studio to demonstrate your program please ensure that you arrive in time to start the development environment.

# Marking Scheme

This is how the marks break down, and how the marks are awarded. Full details, along with the marking sheet, are available from the ACW2 Specification.

## Program Works Correctly – 40%

The game must have these features to get the 40% marks. Missing features will reduce this mark.

- Player number selection and name storage
- Player movement
- “Cheese Power” behaviour
- Game end detection and new game start

## Evidence of Good Design – 10%

A single class is perfectly acceptable as long as variables and methods are appropriately named and used.

- Breakdown into appropriate methods
- Effective data storage

## Appropriate Program Enhancements – 20%

Other enhancements are also acceptable. Note that it is not possible to get more than 20% marks for enhancements.

- “Six Power” – 10%
- Computer Players – 15%
- Board Display – 15%
- XNA Implementation

## Appropriate Documentation – 10%

The user manual should show how to start the program. There is no upper limit on size, although manuals larger than 5 pages should have a reduction in marks. The user manual should contain a screenshot for maximum marks.

- Starting the program
- Playing the game
- Error conditions/reports

## Evidence of Appropriate Testing – 10%

The test report should contain a numbered list of itemised tests and outcomes. At least 10 tests are required for full marks.

## Appropriate Layout and Identifier Selection – 10%

Variable names should match the content. The program should be properly indented with no spurious blocks. Methods and variables should be appropriately commented.

## Testing

There must be provision for testing your program. It should be possible to enter dice throw values for each player. The testing sequence is as follows:

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).
5. A rolls a 2, follows the arrow on (0,2) and lands on square (2,2).
6. B rolls a 2, follows the arrow on (1,2) and lands on square (3,2).
7. C rolls a 6, follows the arrow on (6,0) and moves to (6,6).
8. D rolls a 5, follows the arrow on (0,3) and moves to (5,3).
9. A rolls a 5, follows the arrow on (2,2) and lands on square (2,7).
10. B rolls a 6 and is unable to move because this would take her off the board.
11. C rolls a 6, follows the arrow on (6,6) and moves to (0,6).
12. D rolls a 5 and is unable to move because this would take him off the board.
13. A rolls a 5 and wins.

The dice sequence is:

**2,2,3,4,3, 2,2,6,5, 5,6,6,5, 5**

## Demonstration Sequence

1. Run the game for four players and check that it follows the gameplay sequence as above.
3. Ask for and view all enhancements.
4. Open up the code and mark the design and layout sections.
5. View the User Documentation
6. View the Test Documentation
7. Final discussion
8. Assign mark

Rob Miles 6/12/13