|  |  |  |  |
| --- | --- | --- | --- |
| **Coventry University**  **FACULITY OF Engineering and Computing**  **Coursework Cover Sheet**  *Please ensure that you complete all relevant sections legibly*  ***First Copy:*** *Attach top copy to the front of your assignment.*  ***Second Copy:*** *Keep safety as your receipt* | | | |
| **Module Code**  **210CT** | | **Student Card ID Number**  Please print in BLOCK CAPITALS  Surname…………………LIN.……………………………  Other names………………ZIQIAO….………..........  Signature………………….……...…..…...  **6826257** | |
| **Module Title**  **Programming, Algorithms and Data Structures** | | | |
| **Deadline date**  **Part 1 – 27th Oct 2015**  **Part 2 – 17th Nov 2015** | **Actual word**  **Count N.A.** | | **Tutor**  **CY Cheng** |
| **Coursework Title/Number**  **Assignment** | | | |

This assessment is all

my own work and has not

been copied in part or in whole

from any other source, except for any

clearly marked up quotations. It

complies with the university regulations

on plagiarism, which I have read

and understood.

|  |  |  |
| --- | --- | --- |
| **FEEDBACK ON MARKED WORK:** *Lecturers will complete this section when work is marked.*  *NB. All marks notified during the year are provisional until confirmed by the end of year Assessment Board* | | |
| STRENGTHS  WEAKNESSES  ADVICE ON HOW WORK COULD BE IMPROVED AND FURTHER COMMENTS | | |
| If you require more feedback, please contact your tutor or see module web.  See assignment sheet for assessment criteria for this assignment. | **MARKED AWARDED** |  |
| Less any late penalty | **-** |
| Adjusted mark if penalty |  |

**Marker’s Signature** …………………………………………………………… **Date** ……………………………………………………….

|  |
| --- |
| Students are reminded that reference must be given for any previously published  work used to gather information to help write assignments, including internet  sources, but these sources should not be copied directly. |

**Second Marker Additional comments Signature** ………………Nil…………………………… **Date** …………………………

CLASS DIAGRAM

|  |
| --- |
| **Game** |
| - null\_char : char  - grid\_width : int  - undo\_list\_size : int  - randomFillChar\_array\_size : int  - alphabet : char[]  - randomFillChar : char[]  - clone\_randomFillChar : char []  - grid : char[][]  - clone\_grid : char[][]  - score : int  - clone\_score : int  - counter : int  - clone\_counter : int  - undo\_grid : ArrayQueue  - undo\_score : ArrayQueue  - undo\_randomFillChar : ArrayQueue  - undo\_randomPosition : ArrayQueue  - gameover : boolean  - victory : boolean |
| + Game()  + display() : void  + slideUp() : void  + slideDown() : void  + slideLeft() : void  + slideRight() : void  + randomFill() : void  + gameover() : boolean  + undo() : void  + reset() : void  + gridIsEmpty() : boolean  + gridIsFull() : boolean  + init() : void  + convertLetter(char) : int  + mergeTile(char) : char  + cloneState () : void  + moveValidate() : boolean  + addToUndoList() : void  + generateCharForRandomFill() : void  + shuffle(char[]) : void  + start() : void |

|  |
| --- |
| **Start** |
|  |
| + main(String[]) : void |

<<use>>

|  |
| --- |
| **ArrayQueue** |
|  |
| - maxSize : int  - undo\_list : Object[]  - front : int  - rear : int  - count : int |
| + ArrayQueue(int)  + ArrayQueue()  + empty() : boolean  + full() : boolean  + length() : int  + enqueue(Object) : void  + dequeue() : Object  + pop() : Object  + toString() : String |

<<use>>

|  |
| --- |
| **RuntimeException** |
|  |
|  |

<<use>>

|  |
| --- |
| **EmptyQueueException** |
|  |
| + EmptyQueueException () |

|  |
| --- |
| **QueueFullException** |
|  |
| + QueueFullException () |