

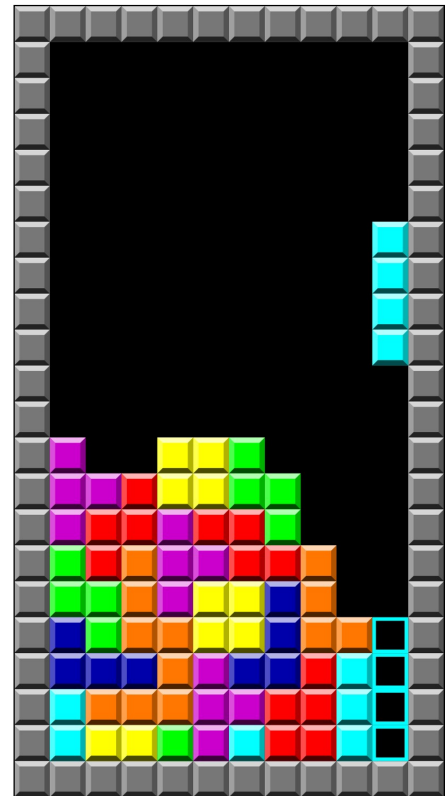
2805ICT System and Software Design
3815ICT Software Engineering
7805ICT Principles of Software Engineering

The purpose of this assignment is to construct and document a Requirements Specification, Design, and Development Process for a classic computer game Tetris. You should apply, as far as possible, the software engineering techniques introduced in this course. You should employ an OO design and demonstrate capability to use advanced design patterns and tactics in your project. The following problem statement is provided. The assignment is to be conducted in groups with no more than four students in one group.

Tetris

“**Tetris** (Russian: Тетрис [['tiɐtrɪs](#)] or [['tɐtrɪs](#)]) is a puzzle video game created by Soviet software engineer Alexey Pajitnov in 1984. It has been published by several companies for multiple platforms, most prominently during a dispute over the appropriation of the rights in the late 1980s. After a significant period of publication by Nintendo, the rights reverted to Pajitnov in 1996, who co-founded the Tetris Company with Henk Rogers to manage licensing.

In *Tetris*, players complete lines by moving differently shaped pieces ([tetrominoes](#)), which descend onto the playing field. The completed lines disappear and grant the player points, and the player can proceed to fill the vacated spaces. The game ends when the playing field is filled. The longer the player can delay this outcome, the higher their score will be. In multiplayer games, players must last longer than their opponents; in certain versions, players can inflict penalties on opponents by completing a significant number of lines. Some versions add variations on the rules, such as three-dimensional displays or a system for reserving pieces.”¹



General description

Apart from the requirements from the original game, your program should have a start up page, the start up page should contain:

1. The title of Tetris
2. The year and course code
3. The list of all students in your group
4. An exit button to exit the program
5. A score button to show top 10 players with their scores
6. A configure button. In the configure page, a player can select normal game or game with extension (explain later), and can select the size of the field, which is the area the blocks move and build, the block dropping speeds (levels). A player can also select AI game (explain later).

¹ Extracted from Wikipedia [Tetris - Wikipedia](#) (21/06/2022)

7. A play button to take the player to the game

Game control

During the game play period, the player will use following keys to control the game

1. Left arrow key to move falling block move left.
2. Right arrow key to move falling block move right.
3. Up arrow key to turn the falling block 90 degree clockwise.
4. Down arrow key to speed up dropping speed of the falling block
5. "P" key to pause and resume the game
6. Esc key will bring a dialog box ask if the player want to finish the game. If the user select "Yes" then return to the start page, if "No" continue game.
7. "M" key to turn on/off music and sound effects.

Game scoring

We use a simple scoring for the game. A player or AI can get points only when they eliminate lines. More lines they eliminate through one block dropping, more points they will receive. The mapping between points and number of lines removed in one round is as:

- 1 Lines = 100 points
- 2 Lines = 300 points
- 3 Lines = 600 points
- 4 Lines = 1000 points.

Game extension

The normal game has 7 different blocks and all of them contain 4 squares (as in Figure 1). The extended game will introduce two more types of blocks with 3 squares each as in Figure 2

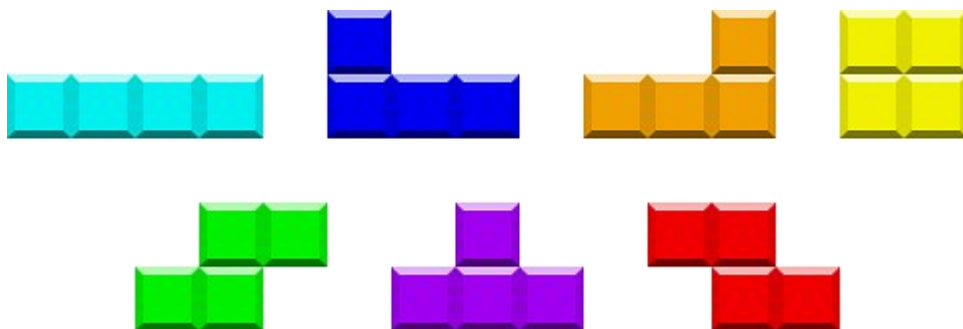


Figure 1. The 7 different shapes of blocks in normal Tetris

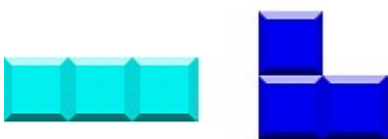


Figure 2. Two more types of blocks introduced in extended game.

AI game

The game has two different play mode: player mode and AI mode. In the player mode, the player uses the keyboard to control the movement of the dropping block as described in Game control section. In the AI mode, game AI will control the movement of the dropping block, to make it move left, right, turn and speedup dropping.

A player can select different mode in the configuration page.

When a player complete a game with score in top 10, the system will ask the player to input a name, and then the name and the score can be found in the top score page.

In the gameplay page, apart from the field of the game, the dropping block, and the accumulated blocks at the bottom part of the field, the page will also need to display other information:

1. Your group number (you will get the number when you enrol in one group).
2. Current score of this play session.
3. Number of lines have been eliminated in this play session.
4. Current level.
5. Extended game or normal game
6. Player mode or AI mode
7. Next block (the shape of next dropping block when current block drops to the bottom)

Game completion

There are two ways for a game to be completed. The first is that when the field is filled with blocks and a new dropping block has no space to appear, then the game is completed. The other way is that during a game, when a player presses ESC key and then click "Yes" in the dialogue box, then the game is also completed.

Whenever a game is complete, if the score is in the top 10, the game will pop up an input dialogue box for the player to input a name. Then the name and score will appear in the top score page.

The rule is applicable for AI mode as well. But no name input is required. AI game will take "AI" as the name.

When a player completes a game with score in top 10, the system will ask the player to input a name, and then the name and the score can be found in the top score page.

MARKING SCHEME - First stage submission

Submission deadline 26 Aug 2022 midnight

Total Marks = 100 = 20% in final grade

REMARKS

The following marking scheme will be used as a guide for marking this assignment. However the assessors marking the assignment may use their discretion where appropriate to deviate from this scheme.

YOU MUST INCLUDE ON THE FRONT PAGE, A LIST OF ALL GROUP MEMBERS (IN ALPHABETICAL ORDER) AND THEIR STUDENT NUMBERS, THE COURSE CODE, FAILURE TO MEET ALL THESE REQUIREMENTS WILL RESULT IN AN IMMEDIATE LOSS OF 10 MARKS.

Try to keep your overall report as concise as possible by appropriate use of graphic and tabular forms and other structuring devices.

This submission contains two elements: a technique report and a video.

Technique report making scheme

Project Planning and Documentation = 10

| Task | Task description | Marks |
|------------------------|--|-------|
| Projection plan | Based on the template to create a project plan <ol style="list-style-type: none">1. Project plan is reasonable and complete.2. The actual working hours are recorded fully.3. The effort and contribution table is complete with justification (if needed) | 5 |
| Version control system | The overall documentation is good regarding layout, spelling, grammar. Use screenshots to demonstrate that a version control system has been applied for the project. | 5 |

Requirements Analysis = 50

| Task | Task description | Marks |
|--|---|-------|
| Functional requirements | All functional requirements are identified, list, and described properly in a table. Each functional requirement should have a unique functional requirement id, and it should be described clearly and completely. | 10 |
| Non-functional requirements | Applying FURPS+ to identify, list, and describe at least 5 non-functional requirements. | 5 |
| Use case diagram | Draw a use case diagram to show all use cases. <ol style="list-style-type: none">1. All the use case diagram elements are included correctly.2. All functional requirements are covered3. All use cases are named properly (as verb + noun)4. Correctly use include and extend relationship | 10 |
| Full use case description | Pick up a use case and develop a complete full use case description. All the elements in a full use case description should be presented. | 10 |
| Requirement - use case traceability matrix | To create a traceability matrix to show the relationship between functional requirements and use cases | 5 |
| Activity diagram | Draw an activity diagram to show the game play procedure from start the program, play the game, finish the game, and exit the program. You need to: <ol style="list-style-type: none">1. Correctly use activity diagram notation.2. Start and stop symbol must be presented correctly.3. At least a pair of synchronization bars are used properly and correctly in one activity diagram.4. At least one decision diamond is used properly and correctly in one activity diagram | 10 |

Video marking scheme = 40

| Task | Task description | Marks |
|----------------|---|-------|
| Video length | You are asked to submit a video to demonstrate the execution of the prototype of the game. The video should be shorter than 1 minute, video content beyond 1 minute will not be assessed. 1. The video should be shorter than 1 minute. | 5 |
| Cross platform | The video demonstrate that the game prototype can execute at least on two different platforms | 10 |
| Start up page | When the game is start, it should show a start up page, on this page it needs to display: 1. The title of Tetris 2. The year and course code 3. The list of all students in your group 4. An exit button to exit the program 5. A score button to get into top score page 6. A configure button get into configure page. Click the exit button will exit the prototype | 5 |
| Top score page | The game prototype should demonstrate following function: 1. To click the score button in the start up page pop up the top score page 2. The top score page will display 10 best scores and the users (The data can be fake in the prototype) 3. A close button in the page, when player click that button, the top score page will be closed and return to start up page | 5 |
| Configure Page | The game prototype should demonstrate following function: 1. To click the configure button in the start up page pop up the configure page. 2. The configure page shows following configure items. (in the prototype, you only need to display following settings, functions of those settings are not asked) a. Size of the field b. Game level c. Normal or extended game d. Player or AI game mode 3. A close button in the page, when player click that button, the top score page will be closed and return to start up page | 5 |
| Game page | The game prototype should demonstrate following function: 1. To click the play button in the start up page pop up the game play page. 2. The game page should display following items a. Game field and a dropping block b. Your group number. c. Current score of this play session. d. Number of lines have been eliminated in this play session. e. Current level. f. Extended game or normal game g. Player mode or AI mode h. Next block (the shape of next dropping block when current block drops to the bottom) 3. The dropping block is dropping. 4. A player can move the dropping block left, right and turn. When the dropping block reach the bottom of the field, it will stop. Other features will not required. 5. Press Esc key will bring a dialog box ask if you want to finish the game. If click "Yes" then return to the start up page, if "No" continue game. | 10 |

MARKING SCHEME - Second stage submission

Submission deadline 23 Sep 2022 midnight

Total Marks = 100 = 20% in final grade

REMARKS

The following marking scheme will be used as a guide for marking this assignment. However the assessors marking the assignment may use their discretion where appropriate to deviate from this scheme.

YOU MUST INCLUDE ON THE FRONT PAGE, A LIST OF ALL GROUP MEMBERS (IN ALPHABETICAL ORDER) AND THEIR STUDENT NUMBERS, THE COURSE CODE, FAILURE TO MEET ALL THESE REQUIREMENTS WILL RESULT IN AN IMMEDIATE LOSS OF 10 MARKS.

Try to keep your overall report as concise as possible by appropriate use of graphic and tabular forms and other structuring devices.

This submission contains two elements: a technique report and source code.

Technique report making scheme

Project Planning and Documentation = 10

| Task | Task description | Marks |
|-----------------|--|-------|
| Projection plan | Based on the template to create a project plan <ol style="list-style-type: none">1. Project plan is reasonable and complete.2. The actual working hours are recorded fully.3. The effort and contribution table is complete with justification (if needed) | 5 |
| Progress report | List the progress in percentage of all functional requirements and use cases. 0% means not start yet, 100% means completed, and other number means under progress. You are expected to complete at least 50% of functional requirements and use cases. | 5 |

Design and software architecture = 60

| Task | Task description | Marks |
|------------------------------|---|-------|
| Class diagram | Draw a complete class diagram to demonstrate your design <ol style="list-style-type: none">1. All functional requirements are covered, all classes are identified.2. Important attributes and methods should be given in each class.3. Correctly use UML notations.4. Should include correct multiplicity5. Should include all three different types of relationship: aggregation (composition), generalization, and association6. Use CapitalCamelCase for class names and lowerCamelCase for method names. | 15 |
| Sequence diagram | Draw a sequence diagram to show one scenario <ol style="list-style-type: none">1. Correctly use sequence diagram notation.2. The class names and method names must match what in the class diagram.3. At least one option frame or if-else frame is used properly and correctly in one sequence diagram.4. At least one loop frame is used properly and correctly in one sequence diagram. | 5 |
| Component and Connector View | Draw software architecture of the system (or part of the system) in a C&C view type: <ol style="list-style-type: none">1. You need to explain the style of this view.2. The view should cover the entire system3. The view should be a high level enough to highlight the structure. | 5 |
| Deployment | Present the project's software architecture in deployment styles | 5 |

| | | |
|---|---|----|
| style view | <ol style="list-style-type: none"> 1. You need to create a diagram to show the deployment setting, both hardware and software should be considered. 2. You also need to give a list view to show the deployment environment of the project | |
| MVC architecture design pattern | <p>You need to apply MVC architectural design pattern in your game. You need to</p> <ol style="list-style-type: none"> 1. Briefly explain MVC 2. Use design diagrams (part of a class diagram and or sequence diagrams) to explain how MVC implied in your design. 3. Use relevant source code (copy relevant part of your source code) to show how MVC has been applied in implementation. 4. Make sure the design diagrams and the source code are matching each other. | 10 |
| Two other design patterns (10 pts each) | <p>You need to demonstrate that you've applied two design patterns (not including architectural design patterns such as MVC and three tiers) in your game. For each design pattern, you need:</p> <ol style="list-style-type: none"> 1. Explain what design pattern has been applied and why to use it. 2. Use design diagrams to demonstrate your design 3. Use relevant source code to demonstrate your implementation 4. Make sure the code and the diagram are matching each other. | 20 |

Source code making scheme

You need to put all source code files in a ZIP file. You ZIP file should only contain source code files and a Readme.txt, no other resource files such as images and soundtracks. If you use some third-party library, you don't need to put them in the ZIP file. Your Readme.txt file will summarize your source code.

Source code section = 30

| Task | Task description | Marks |
|-------------------|--|-------|
| Readme file | <p>You should have a readme file in text format in your source package. The readme file should contain following information:</p> <ol style="list-style-type: none"> 1. List of source code file, and the explanation of each file. 2. Number of lines of each source code file and the total number of lines. 3. Naming convention of classes, objects, functions and variables. | 10 |
| Source code files | <p>The assessor will check the source code if they are:</p> <ol style="list-style-type: none"> 1. Original (not copied from other people's work). This is essential. 2. Matching your design. 3. At least 50% of functional requirements should be covered by source code at this stage. | 10 |
| Naming convention | The assessor will check the source code if they are following a consistent naming convention described in the readme file | 5 |
| Suitable comments | The assessor will check the source code if they have suitable comments for functions, classes, important variables and complex operations | 5 |

MARKING SCHEME - Final submission

Submission deadline 14 Oct 2021 midnight

Total Marks = 100 = 30% in final grade

REMARKS

The following marking scheme will be used as a guide for marking this assignment. However the assessors marking the assignment may use their discretion where appropriate to deviate from this scheme.

YOU MUST INCLUDE ON THE FRONT PAGE, A LIST OF ALL GROUP MEMBERS (IN ALPHABETICAL ORDER) AND THEIR STUDENT NUMBERS, THE COURSE CODE, FAILURE TO MEET ALL THESE REQUIREMENTS WILL RESULT IN AN IMMEDIATE LOSS OF 10 MARKS.

Try to keep your overall report as concise as possible by appropriate use of graphic and tabular forms and other structuring devices.

This submission contains two elements: a technique report and a video.

Technique report making scheme

Project Planning and Documentation = 10

| Task | Task description | Marks |
|------------------------------|--|-------|
| Projection plan | Based on the template to create a project plan <ol style="list-style-type: none">1. Project plan is reasonable and complete.2. The actual working hours are recorded fully.3. The effort and contribution table is complete with justification (if needed) | 5 |
| Automate document generation | The overall documentation is good regarding layout, spelling, grammar etc. Use screenshots to demonstrate that doxygen or other auto documentation generator has been used for the project to generate documents. | 5 |

Advanced Design = 20

| Task | Task description | Marks |
|---------------|--|-------|
| Design tactic | You need to demonstrate that you've applied one design tactic to improve certain quality attributes: <ol style="list-style-type: none">1. Explain what design tactics has been applied and what quality attributes can be improved through the design tactics.2. Use design diagrams to demonstrate your design3. Use relevant source code to demonstrate your implementation4. Make sure the code and the diagram are matching each other. | 10 |
| AI algorithm | Study the web page (Coding a Tetris AI using a Genetic Algorithm – Lucky's Notes (wordpress.com)) or other web pages to find a simple AI algorithm to play Tetris. You need to <ol style="list-style-type: none">1. Explain the algorithm in natural language.2. Describe your implementation in suitable diagrams3. Write pseudocode for this algorithm.4. Show relevant code in your program. | 10 |

Testing = 10

| Task | Task description | Marks |
|---------------------------|---|-------|
| Software test description | Based on the STD template, produce a software test description. The STD is part of your submitted report, you don't need to submit extra document. Your STD should contain: <ol style="list-style-type: none">1. Test environment: detailed description of the operating system, hardware, and required software.2. Test cases: minimum three test cases based on test case template from the lecture or lab exercise. | 5 |

| | | |
|----------------------|---|---|
| Software test report | Based on the STR template, produce a software test report. The STR is part of your submitted report, you don't need to submit extra document. Your STR should contain: <ol style="list-style-type: none"> 1. Test results: all the test cases list in your STD must be excuted and show the test result. 2. Summary table for your entire test activities. The summery table should contain number of errors, type of errors. | 5 |
|----------------------|---|---|

Reflection = 0

| | |
|---|---|
| Master students need to write one page project reflection, fail to do so or the reflection doesn't make sense could result up to 10 mark deduction. | 0 |
|---|---|

Video marking scheme = 60

| Task | Task description | Marks |
|-------------------------|--|-------|
| Video length | You are asked to submit a video to demonstrate the execution of the full game. The video should be shorter than 3 minutes, video content beyond limit will not be assessed. <ol style="list-style-type: none"> 1. The video should be shorter than 3 minutes. | 5 |
| Normal game functions | All normal game functions are demonstrated, move block left, right, turn, speed drop, eliminate filled lines, update scores etc. | 15 |
| Game completion | Two different ways of game completion are demonstrated | 5 |
| Top score page | The video demonstrates the update of top score page. | 5 |
| Sound and music | The game has suitable sound effect and music (you can pick up whatever sound effects and music) | 5 |
| Extended game | Game with extension demonstrated | 5 |
| AI game | AI game demonstrated | 10 |
| Different levels | Different levels of the game are demonstrated. | 5 |
| Different size of field | Demonstrate different size of the field. | 5 |

You may use following tables as templates for your project management.

| Task | | Plan | | | | Actual | | |
|------|---------------------|------------|--------------|-----------------|---------------|--------|-----------------|---------------|
| # | Task Name | Student | Planned Time | Cumulative Time | Finished Date | Time | Cumulative Time | Finished Date |
| 1 | Project plan | Smith | 2 Hours | 2 Hours | 17/03/20 | 2.5 | 2.5 | 18/3/20 |
| | | Mark | 3 Hours | 3 Hours | 17/03/20 | 3 | 3 | 17/3/20 |
| | | Task Total | 5 Hours | 5 Hours | 17/03/20 | 5.5 | 5.5 | 18/3/20 |
| | | | | | | | | |
| 2 | Documentation | Smith | 3 Hours | 5 Hours | 21/03/20 | 2 | 4.5 | 21/3/20 |
| | | Anna | 2 Hours | 2 Hours | 20/03/20 | 3 | 3 | 21/03/20 |
| | | Task Total | 5 Hours | 10 Hours | | 5 | 10.5 | |
| 3 | Draw Behavior Trees | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |

Project Working Load Summary

| | Plan (hours) | Actual (hours) |
|--------------------------------|--------------|----------------|
| Total time | 100 | 105 |
| Average time per person | 20 | 21 |
| Smith | 20 | 25 |
| Anna | 20 | 15 |
| Mark | 20 | 20 |
| Joe | 15 | 18 |
| Tom | 25 | 27 |

Individual marking adjustment table (Example)

| Student | Effort Level* (Rating from 0 – 5, the information is filled by the group) | Contribution Level* (Rating from 0 – 5, the information is filled by the group) | Justification If a student received level rating of 3 or less, your group need to give explanation for the low level rating |
|--------------|--|--|--|
| Smith | 5 | 4 | |
| Anna | 5 | 5 | |
| Mark | 5 | 5 | |
| Joe | 4 | 3 | Joe didn't complete all the tasks assigned to him (he didn't draw the class diagram assigned to him) and Mark helped to complete the tasks.... |

- *Level ratings, 5 = excellent, 4 = good, 3 = reasonable, 2 = poor, 1 = unacceptable, 0 = none

Individual score is calculated based on the score of the submission and the student's effort level and contribution level. Suppose the submission score is S, a student's effort's level is e and contribution level is c. The student's score is calculated based on the algorithm below:

$$T = e + c$$

| T | Student score |
|------|---------------|
| 7-10 | S x 100% |
| 5-6 | S x 80% |
| 3-4 | S x 40% |
| 1-2 | S x 20% |
| 0 | 0 |