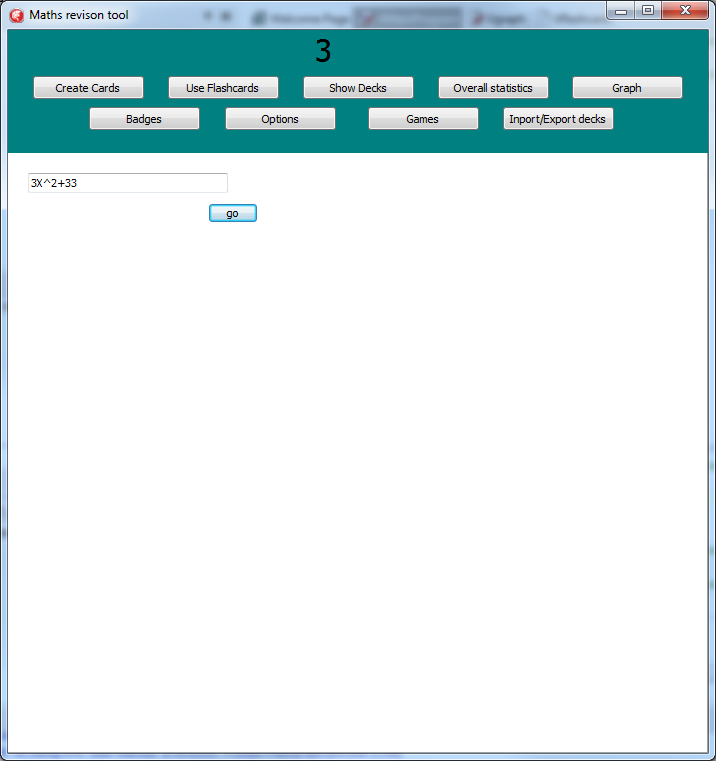
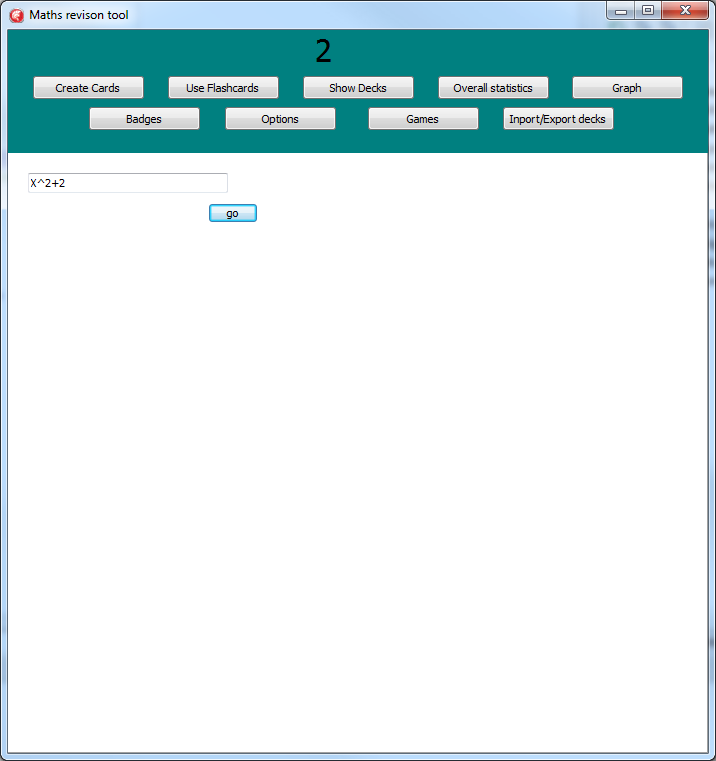
System Testing

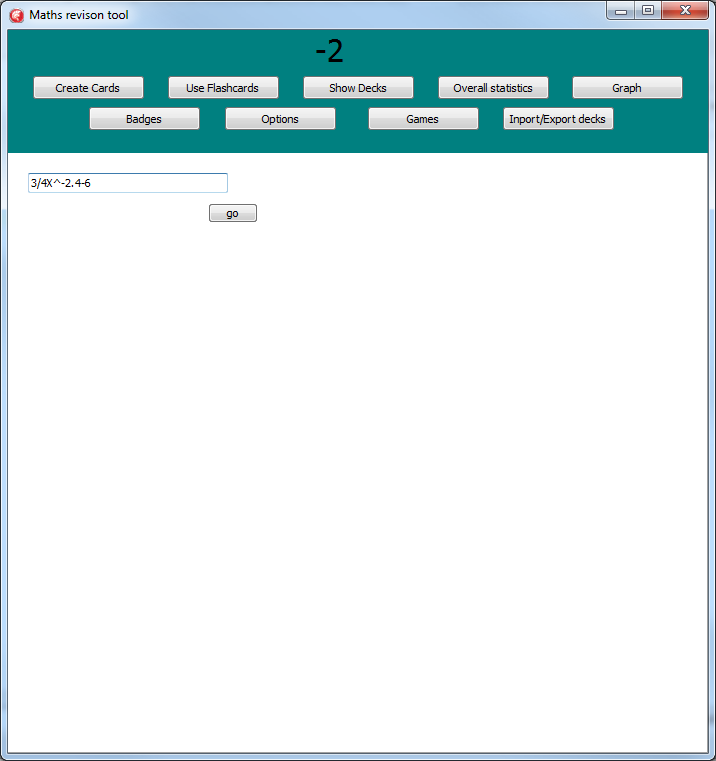
# Overall Test Strategy

As I was programming my project, I would run black box unit testing on new algorithms and methods with a range of ordinary, boundary and incorrect cases to see what the result was, and if a module was particularly essential and I would run a bottom up test on just that process to check it worked before creating other processed that used it. This testing method involved choosing a variety of input data and seeing if the correct output was produced.

Eg. Tests of functions needed for graphing, without creating a graph - showing a bottom up test, as these were some of the essential processes.



The first shows testing for an integer in an equation, and the second shows testing for a coefficient, both of these were successful, however not all my tests were;



This test that was looking for the power X was raised to failed.

When it failed the black box testing I had to use unit white box testing and follow the algorithm through to find where an error occurred, or what caused as early/late exit to a procedure or loop. In some occasions this required dry run tests and I had to see what caused a divide by zero, infinite loop , or invalid pointer operation. Sometimes this made me realise my initial algorithm wasn't good enough so I had to rewrite the algorithm/data structure so it worked with the new circumstance. Once I'd found the error I would correct it and retest that section using a black box regression test, to ensure the problem had been fixed, and that nothing else had broken.

I also had to do lots of integrating tests to check that my database and program functioned properly and the correct data got inputted/ queried when needed. These tests were also done to check the objects within my program worked with each other as I intended. They were also essential when I created my graph as it required lots of smaller procedures working in different orders or under different circumstances for each equation it had to draw.

# Test Plan summarised

1. After writing a new section, and if it was possible to run alone then run black box testing on it, which where I checked the outputs were correct for the tested inputs.
2. If no errors were found then see how it works with other aspects of your program, if any errors are found then pinpoint them using white box testing and dry runs. By knowing where the problem occurred and what caused it I can edit/ rewrite the algorithm to amend this.
3. Retest the fixed section.
4. Continue testing new sections as I make them.
5. Test how the system works overall and whether it meets the specification.
6. Get User to test the program and see what they have to say.

# Test Sections within my Program

All Test Results can be seen below

## Graph

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Test Data/Action | Reason/Explanation | Expected Result | | Actual Result | Test Type |
| Test 1 | Testing y=1/x on the main graph | Tests how the graph can handle dividing by zero, and drawing a basic line, also a coefficient is needed to be found without one directly written, however the characters that are in front of the X are '1/' which are both valid characters for a real in my string, so I need to test that it doesn't think that is the coefficient and +1 is instead. | | No errors, Line drawn is a regular 1/x curve. | As Expected | Boundary as 1/x isn't a usual polynomial form.  /Processing/Valid |
| Equation had been inputted as a string and decoded into a graph.  No error occurred when x=0  Passes through (1,1) which shows it’s a 1/x curve  Correct line drawn | | | | | | |
| Test 2 | 2/3y+2.6=(x+1)^2.3  On regular graph section. | Tests how the y coefficient is gathered, and how polynomials in brackets are handled.  The real power is to test what happens to negative x values as this doesn't exist. | Graph below (Desmos) | | As expected | Boundary/Valid/ processing |
| No graphs beyond this point, meaning real powers don’t produce errors when X is negative.  Same shape as the online graph, which shows that Brackets to a power work and that Y coefficients are correctly obtained. | | | | | | |
| Test 3 | Y=c(x) on regular graph section. | No function for this exists, and should test whether it things this is cos/cosec but then reject it. | No graph is drawn, as the equation is invalid | | As Expected | Invalid/boundary |
| No graph has been drawn, and no error was produced for the incorrectly inputted function. | | | | | | |
| Test 4 | Y=1/2x on a graph flashcard | Checks that o.5X is plotted not 1/(2x) or 1/(1/2x), and tests if a graph flash card works properly. | A linear graph with gradient 1/2 is drawn is shown on the flashcard. | | As expected | Boundary/Valid/ Processing / UI |
| Graph is linear at the correct gradient showing the coefficient was correctly obtained.  Equations inputted on a flashcard correctly create a graph on a card. | | | | | | |
| Test 5 | Y=1/x+1 , on a regular graph. | Checks that the order of operations on the equation only divides x by one, then adds one onto it. | The graph shown in test one is displayed but the line is 1 unit higher. | | As expected |  |
| The horizontal asymptote is now at +1 not 0 , which shows the graph has correctly translated up one unit.  This shows that y=1/(x+1) wasn’t graphed as this would have moved left one unit.  It also shows that an integer on its own in the equation is correctly obtained. | | | | | | |
| Test 6 | Y=-3/7E^(x) on a regular graph | Checks if negative numbers work, and if fractional coefficients work. | A slightly shorter y=e^x graph . but is reflected in the x axis. | | As expected | Boundary / Valid / Processing |
| Graph passes above (0,-1) , which shows that the coefficient of -3/7 has correctly been drawn and that fractional numbers work correctly.  Graph is below the X axis, this shows that a negative coefficient correctly reflects the normal graph in the x axis. | | | | | | |
|  | | | | | | |
| Test 7 | Y=-x , on a regular graph | Boundary test as there is no coefficient but one is needed. | A y=-x graph is drawn on the graph with no errors. | | As expected | Boundary / valid / processing |
| Graph is correctly drawn with a gradient of -1, showing the correct coefficient is produced. | | | | | | |

## Graph Initial tests on small processes

## Randomly Generated Questions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No. | Test Data/Action | Reason/Explanation | Expected Result | Actual Result | Test Type |
| Test 8 | Check that a pack with one card will continue to work. | As the algorithm makes sure last cards aren’t selected, but if it’s the only choice for card it needs to be selected. | Same card appears over and over. Card ratings updates, and so does cards seen. | As expected | Boundary |
| Database showing that the pack being chosen only has one card, and that cards seen is 0.  Above is the card data, showing the options, correct answer, and question.  The only cardID linked to the chose pack.  The deck being chosen  Above shows the card generating correctly the first time with the correct graphs, question and answer.  When the rating is chosen the card below is generated, and as it’s the same card due to it being the only card in that pack it shows this test weas successful.    Below shows the Card with its updated ratings after being seen once.    Below shows the cards seen has increased by one as expected, and that the ratings have updated. | | | | | |
| 2 | Check a level two card is created in a deck with 7 cards at above 80%. | As this tests that a level 2 card is created at the boundary between level 1 and 2. |  |  | Processing/ Boundary |
| 3 |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Menu Screen

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Test Data/Action | Reason/Explanation | Expected Result | Actual Result | Test Type |
| Test 9 | Opening it for each option. | Each option should show different stats. | New card for 'create' , favourite for 'run' and 'view' and nothing for copy | As expected | Valid |
| New deck is shown for create as expected Favourite is shown for Run as expected    Favourite is also shown for view. No extra panels shown for copy as expected. | | | | | |
| Test 10 | Selecting Multiple decks in flashcards. | As this shouldn’t be allowed. | Error message saying you cant | As expected | Invalid |
| Correct error message shown when user clicks continue.  Multiple decks being selected | | | | | |
| Test 11 | Selecting Multiple Packs(not Fav) in flashcards. | As this is allowed, other than when creating a card, or selecting fav/new card. | If it’s not creating a card, then proceed to either View or run the multiple packs | As expected | Boundary |
| Two packs being selected.  No error is shown, the program correctly continued to run flashcards. | | | | | |
| Test 12 | Selecting multiple packs with Fav/New in decks. | As this is the same as before but not allowed. | Error message saying that you can’t select both. | As expected | Invalid |
| Two packs including favourites being selected.  Correct error message shown | | | | | |
| Test 13 | Clicking overall stats button at the top. | Check that the user can exit out of the menu. | Menu closes, and new window opens. | As expected | Valid/UI |
| The previous screen has completely gone, and has been replaced with the correct stats screen. | | | | | |

## Create a card

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Test Data/Action | Reason/Explanation | Expected Result | Actual Result | Test Type |
| Test 14 | Creating a regular card in a new deck called Deck, and then a new pack called pack. | As it needs to open other windows first and nothing exists in the database for these packs yet. | Opens create deck window, then shows deck and create pack, then create card. Database has new deck and pack added with 0 stats, and card is added with test data. | As expected | Valid/UI/DB |
| Database showing no deck called deck existing  Selecting to create a new deck    Below shows new deck in database, with 0 stats  New pack being selected to be made  New deck is now shown on menu  Creating the deck with correct name    New pack created with 0 stats  Pack being created  Basic create a card screen is shown.  Field in PackDeck correctly created.  Question and answer for the new card.  Window changed to correct menu for a basic card.  Below shows the database before the card has been added.    Below shows the database after the card has been added, the question is in the first column, answer in the second. The stats have been set to zero and the favourite is false. This is as the data was inputted showing the test was a success. | | | | | |
| Test 15 | Creating a card in new deck at deck limit of 5. | As it shouldn't let you create a new deck. | Error message telling you you're at deck limit and menu remains. | As expected | Invalid |
| Correct error message shown and menu hasn’t changed.  Selecting new deck what 5 decks exist. | | | | | |
| Test 16 | Creating a card In multiple packs | As this isn't allowed. | Error message telling you that you can't add to two packs. | As expected | Invalid |
| Selecting 2 packs  Correct error message shown | | | | | |
| Test 17 | Entering no data to a new options graph card | The program needs data to send to the database, if no data was present issues would happen later. | Error message telling you what data is missing. | As expected | Invalid / Boundary /exceptional data |
| Error message telling you everything that is missing, correctly displayed and screen behind hasn’t changed.  Empty Data in all boxes of the menu. | | | | | |
| Test 18 | Selecting an incorrect option | This would mean an option with no text and not sent to the database is the correct answer, making the card impossible. | Error Message telling you the correct option is incorrect, menu isn’t cleared. | As expected | Invalid / Boundary |
| Error box is correctly shown saying that just the correct answer is missing, as all the other inputs are correct. The menu behind correctly doesn’t change.  Option 6 has been selected as the correct answer  Options 1-3 Have text | | | | | |
| Test 19 | Filing in options in the wrong order | As entering those in the wrong order would put empty answers into the database. | No error message, Proceeds to add all the inputted data and not the empty boxes into the database. | As expected | Boundary / exceptional data |
| Fav has been picked.  Question has been inputted  Options haven’t been in putted in the first 3 boxes, but a valid option has been chosen as correct.  Below shows the database at the start, without the card data that is being inputted.    Below shows the inserted field into the database, which is correctly inputted with the question in the first column, options at the last 3 in consecutive option numbers , with the last one (3)being correct as opposed to 6 which was inputted. This shows the test was a success and the data was inputted correctly. | | | | | |
| Test 20 | Creating an options card, with 6 options, that is favourite with option six being correct. | Is a standard input that is expected and should have no errors, 6 options is the max so it's a boundary test. | Creates New card in database with the data. | As expected | Valid / normal data /boundary |
| Card is an options card, with all 6 options and six is correct.  Correctly inputted field into the database shown below, before state is seen the last test. | | | | | |

## Running Flashcards

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Test Data/Action | Reason/Explanation | Expected Result | Actual Result | Test Type |
| Test 21 | Choosing Deck Fav | As this requires only favourite cards to be selected. | You will only see favourite cards | As expected | Valid/DB processing |
| Fav is selected  Only this card is a fav in the database, meaning the test was a success  Star is gold showing its fav | | | | | |
| Test 22 | Multiple Packs with each having one card | Both packs cards need to be selected. | Two cards will be seen when running the packs. | As expected | Valid/Db Processing |
| Second card is shown afterwards  First card being shown from one pack  Multiple packs have been selected  Packs selected only have one card in the database | | | | | |
| Test 23 | Clicking Edit Card on an options card. | As this needs to interact with the object to get data, then the db to submit it. | Opens up the create a card window with the data already filled in, editing the data changed the flashcard and the database. | As expected | Valid/DB processing |
| Edit menu correctly opened with all the data from the card in the correct boxes.  Has 6 options  Edit button being clicked.  Below shows the cards field in the database , with that same information as shown above.    Upon edit click, the data has updated on the flashcard on the screen  Question, Options, correct option, and fav have all been changed.  Below shows the editied data in the database, which has been correctly updated. | | | | | |
| Test 24 | Clicking Delete Card when only one card is available. | As it needs to stop running the flashcards after this, as there are no remailing cards. | Card is removed from screen, deleted from db, pack number and deck number is decreased. Menu screen is opened. | As expected | Boundary/DB processing |
| Only 1 card in the selected pack and deck  0 cards are now in the deck, and in the pack  Delete being clicked | | | | | |
| Test 25 | Flipping the card, with no answer. | As if an answer is needed, the card shouldn’t flip, and the time shouldn’t stop. | Card doesn’t flip, time doesn’t stop, and screen looks the same. | As expected | Valid/ Boundary |
| When card was clicked nothing changed  Empty answer box | | | | | |
| Test 26 | Updating user rating to 5, on a card with no answer needed, within 8 secs of card opening. | Tests card knowledge calculation | Processing user rating and time to get a card rating of 100%. Which is added to Db, and the pack and deck %’s are updated correctly. Cards seen and stats is also updated. | As expected | Valid/processing |
| Pack selected with one card that hasn’t been seen.  Turned off answers in the options  Below shows the data for the card that’s being loaded, as you can see its stats are all 0.    Below is the pack it’s in with 0 stats again.    Below is the deck it’s in in with zero stats again.    Below is the user over all stats showing that 40 cards have been seen (column 2)    Progress towards correct streak badge is currently at 20  Progress towards cards seen badge is currently at 40  User rating 5 is picked after less than 8 secs.  Pack displayed with correct data and without an answer box.  Pack has been updated to 100% and bar chart now shows all cards area rated 5.  Below shows the cards field in the database with its updated stats, that collumn saying ‘3’ shows the last time which is under 8 secs, the collumn saying ‘100’ is the card rating at the correct value, and the one saying 5 is the user rating showing the number that was selected.    The last three columns in both the deck and the packs fields show that average user rating is now 5 (correct) and that average card raring is 100 is also correct. The cards seen in the last column has also correctly I creased by one.    User stats’ cards seen has correctly increased to 41.    Correct streak stats haven’t increased as an answer wasn’t needed.  Progress towards cards seen has increased to 41 in all unlocked badges, and none have unlocked, which is correct. | | | | | |
| Test 27 | Opening a randomly generated question pack, and seeing a new differentiation card be created into an empty deck. | Boundary test as usually opening an empty pack isn’t allowed, but a new question should be generated. | Card generated in correct format and appears on the screen. Card is added to Db and pack numbers are updated. | As expected | Boundary/db processing |
| Card question and answer is correctly generated with option 5 being the correct answer.  Empty differentiation pack is selected.  Below shows that the differentiation pack now contains one card. | | | | | |

## Viewing Flashcards

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Test Data/Action | Reason/Explanation | Expected Result | Actual Result | Test Type |
| Test 28 | Viewing Favs with one card | Boundary correct value, and should open showing one card. | View card screen is opened with only one card. | As expected | Boundary / Valid /db query |
| Only one card is displayed, and it has the fav star showing it’s favorited.  Favs is selected | | | | | |
| Test 29 | Viewing Fav With 0 cards | Is a boundary incorrect value, and shouldn't run. | Error message telling you there are no cards shows up, and menu remains. | As expected | Boundary/ invalid / db query |
| Correct error message shown, and menu behind doesn’t change.  Selecting fav | | | | | |
| Test 30 | Viewing pack ‘Test 1’ with 4 Cards | It’s a typical situation and checks normal situations, and tests correct number of pages is created. | Opens up a menu with the first two cards in the pack shown, with buttons ‘page 1’ and ‘page 2’ at the bottom. | As expected | Valid/ db processing / Processing / sufficient records |
| There are two page buttons as was needed.  Two cards correctly shown on this page  Four cards are in the selected deck. | | | | | |
| Test 31 | Going to Page 2 (of above) | Tests it goes to the correct cards for that page. | Cards on screen change to the 3rd and 4th card in pack. | As expected | Valid/ db processing / Processing |
| Two other cards are now displayed.  Page two has been clicked | | | | | |
| Test 32 | Deleting Top card (from before) | Checks that the top card is deleted and nothing else. | Top card is deleted from screen and db, card at the bottom moves up to the top, with card 3 now being at the bottom. | As expected | Boundary / Valid / Processing |
| Card 3 (seen in test above) is now at the bottom.  Bottom card has moved up to fill the deleted cards position.  Top card is deleted.  Packs field in database before showing 4 cards.    And after showing 3 cards. | | | | | |
| Test 33 | Deleting Bottom card (from before) | Checks that the bottom card is deleted and nothing else, but the page number also decreases. | Bottom card is deleted from screen and db, card from2nd page fills its space, and ‘page 2’ button disappears, with ‘page 1’ repositioning itself. | As expected | Boundary / Valid / Processing |
| Number of page buttons has correctly decreased to one.  Card no.4 is now in the correct space.  Packs field in database now shows that it has 2 cards instead of 3. | | | | | |
| Test 34 | Viewing a multi choice graph flashcard. | To check this format displays correctly at a different size. | No options buttons for the options, graph and less axis labels, and card is displayed from the data in the database. | As expected | Boundary / Valid |
| axis labels are now in stapes of two not 1.  Options don’t have radio buttons  Graphs show the correct equations  Card, and user ratings are correct and correct colour  Previous answer is correctly displayed in green.  Times are correct  Question is correct | | | | | |

## Badges

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Test Data/Action | Reason/Explanation | Expected Result | Actual Result | Test Type |
| Test 35 | Badge page is opened, with some badges unlocked. | Is a regular process for this section to do. | Page opens showing 18 badges, with the correct information and stats from the database. | As expected | Valid/ Processing / Db Processing |
| Three are unlocked as before.  Three badges have been unlocked and look different.  18 badges are correctly displayed, with the correct stats, names, progress, picture and colour. | | | | | |

# Trace Tables/ Dry runs

# System Testing

My program produces the results that were needed and expected in all of the sections and its meets my specification, and the different modules within my program work with each other and don't cause any crashed if sections are opened in a random order.

# Test evaluation

I believe the way I tested worked very well, as testing it in sections as a black box let me quickly see what worked and get onto testing data that was less likely to work as it was boundary/invalid. This meant I managed to quickly find the problems of my code, and pinpoint where they happened as I was testing after each new section. I could then look at this section, and knowing what circumstance caused it to fail, think of a new solution or fix to the problem. I could then test if the fix worked, didn't cause other parts to break and move on to test other boundary data. This meant I could efficiently check each new section and gradually build my program up piece by piece.

One problem I found with the way I was testing was that I sometimes didn't come across an error until a later stage in the project. This was due to me not having thought of all the situations to test, such as I hadn't thought of certain dividing by 0 errors, and order of operation methods until I drew a graph and it was in the wrong place. This meant I didn't come across all the errors at first, but still managed to catch a lot afterwards as I was building up from the sections more and more.