

# Bridge HCP Program Testing and Plan

## Task

The task is to create a program to help the Paraparaumu College Bridge players practice their HCP calculation skills and for the program to be easy to use for students. The program must be commented and easily understood in the event that one of the Bridge club operators that also know how to code in python (Mr Brown) can then edit the code if they want to add further functionality later down the line. The program must be able to calculate at least one hand (either north, east, south or west) and should be able to calculate the HCP and let the student calculate what they think the HCP is and match it against the actual HCP, the program should then let the student know if the HCP is correct or not and if not it should let the student know what the correct HCP was along with some help on what the point value of the cards are.

## Programming method

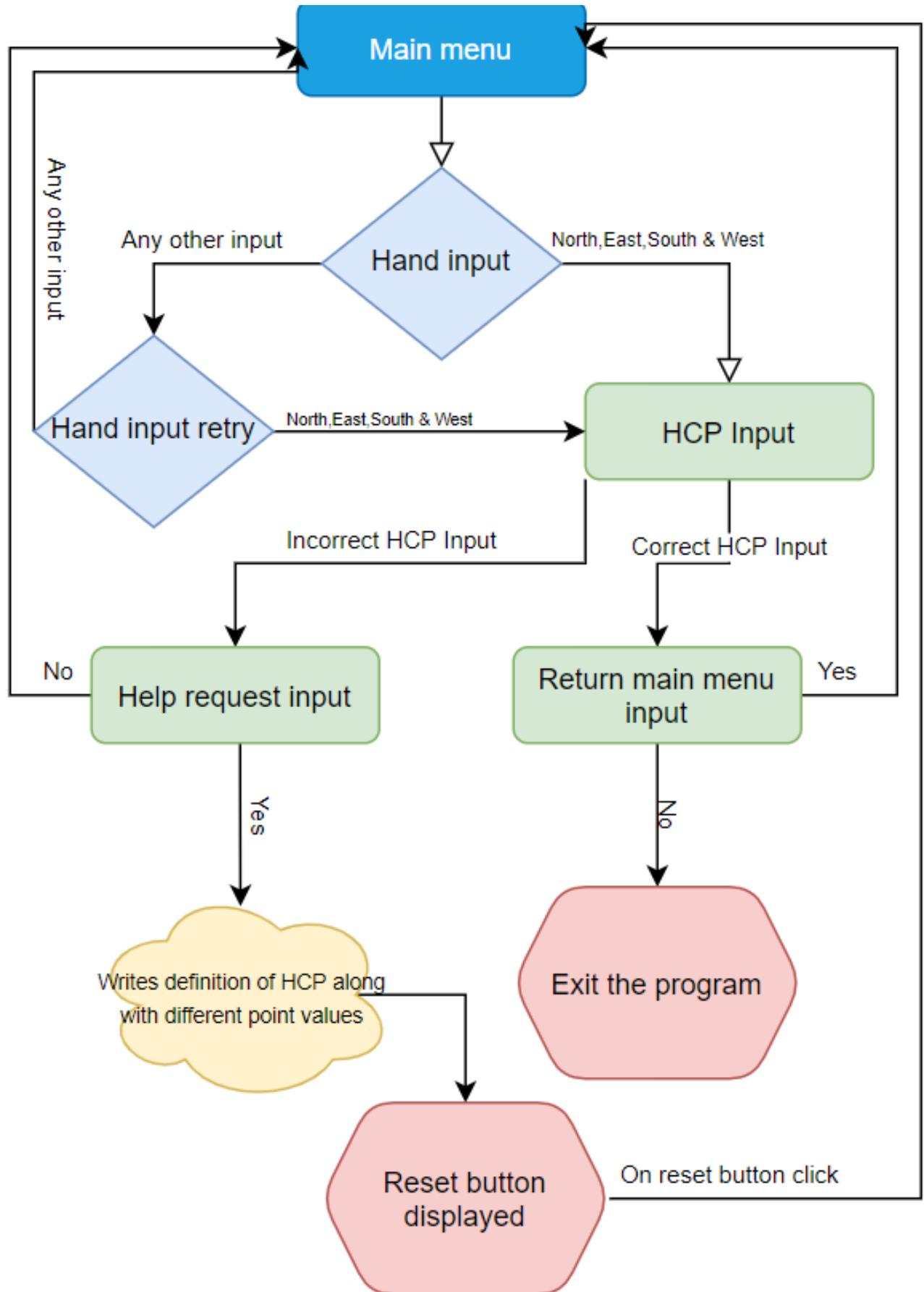
The programming method being used is agile, agile software development is done by using programs such as trello and github to manage your programming, instead of using the outdated “waterfall” method of management where the whole program was planned out before with documentation and other planning, this was seen as a better move back then by the higher up executives and businessmen although didn’t actually end up working and over time the agile method of programming was developed and implemented into the development of many different programs. In my program i will be using tools such as github, trello and repl to manage my program with github giving a place where I can push changes to the main branch or make other branches where more experimental developments can be made to be later merged if the developments turn out to run properly with it being easy to revert back to the master branch if necessary. Another tool being used was “trello” trello boards are essential to project management when it comes to developing a program, if the program is being made by a single developer it helps organize different things and makes sure that you know what to do and what’s done and when it comes to teams using a project management tool such as trello is definitely essential due to issues such as multiple team members coding the same module or function when they should have been well organized and were doing different things. Repl is essentially an IDE but online, repl integrates nicely with github and also has a “google docs” styled editor with auto saving on repl’s servers, this is great because it means that you can access your program from anywhere as long as you have a web browser and you don’t need to constantly worry of saving and backing up your files to a third party as long as you frequently make new pushes to github, repl also grants access to a remote virtual desktop where you can run and control any graphical gui within python to further help simplify things and streamline them.

## Program outline

The purpose of this program is for it to scan in a bri file and identify what cards are in each hand (north, east, south west) and show the user a hand and the cards within that hand and then have the beginner bridge player calculate the HCP (high card point) and tell the user that they were correct or not and how they can improve themselves.

The interface should be a gui that asks the user to enter the HCP from the cards displayed on screen and if they don't answer correctly they get a help prompt, if they do answer correctly they should be able to return to the main menu and try again or instead try a different hand.

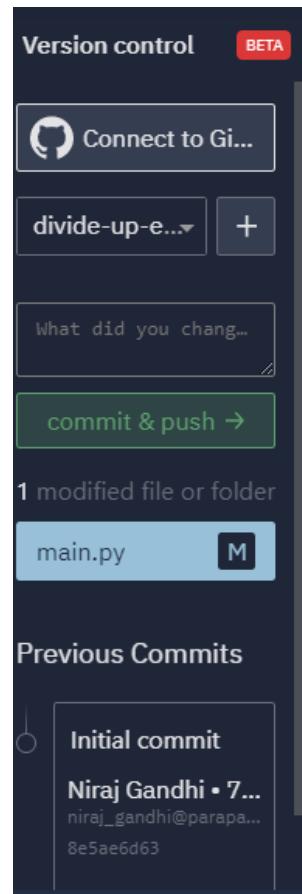
## Plan



Above we can see the GUI input plan which are the series of events that the user will go through and is a plan for the programmer to follow, this plan helps streamline future decisions and overall is good programming practice

#### Relevant implications:

- The program must carry out its task of reading in 13 cards from a .BRI file and should ask the student using the program what the HCP is (High card point) and will tell the student if it's correct or not. The BRI file contains 8 boards worth of cards and the program will also figure out the missing "hand" 13 cards by adding a full set of cards in a list and removing them as it finds them from a BRI file.
- Tools used to help for version control, I will be using GITHUB and will be using it with "repl.it" for version control by making a new branch every time I make a change so I can easily roll back if there are many issues with the code.
- The tool I will use for project management to mark off what I'm doing is Trello which will help me navigate what I'm trying to do and will let me easily manage time and list down what has been completed and what is a work in progress.
- 



#### LOG:

For logging I will be using Trello boards which is a free solution to manage what is needed to be done, in progress and completed.

Progress made	info	date
Creating empty list 1-52	This is to create a predefined list with values 1-52	1/09/2020
Calculate board	The board includes North, East, and South hands and all are calculated	1/11/2020
Calculate west hand	As provided with only	1/12/2020

	North, East and South hands in the encoded bri file, all that needs to be calculated is the fourth hand (south) this can be done by using “sets” and subtracting from a list containing all numbers up to 52, this will leave the west hand calculated.	
Create a database in excel and save as .csv and make this .csv a python dictionary.	Create a basic database that “decodes” the encoding used for the BRI file, meaning that a different BRI file can be used without any issues and the program will operate correctly as the database includes (BRI value,card name,card point value)	1/13/2020
Export information from database into lists for each hand (north, east, south, west)	All lists for (N,E,S,W) include all the point values.	1/13/2020
Calculate the HCP of each hand (N,E,S,W).	This is done by adding the points all up in each hand to find the HCP.	1/14/2020
Start on the GUI	This is done by importing turtle and creating a screen	1/14/2020
Create list with card name	Create a list from the database for each hand that has the card name, the .gif file for every card is also named this, what this ends up doing is allow to display the turtle shape of the card by adding “.gif” to each item in list.	1/15/2020
Create csv file in excel	with the BRI encoding information (what the bri	1/15/2020

	encoded value corresponds to in terms of, points and card name)	
Start on cards deck layout	Have the cards displayed in a nice gui format	1/16/2020
Create all the input screens	All the user input windows that allows the user to enter their commands into	1/16/2020
Make logo in photoshop	Aesthetically pleasing logo	1/17/2020
Create the functions for each prompt	Making sure that the functions all lead to the current places and are error checked	1/17/2020
Create reset button	Reset button where when clicked runs the main() function which essentially restarts the program.	1/18/2020
Polishing of code		1/18/2020
Boundary / corner case checking	This is to ensure the program doesn't get "stuck"	1/19/2020

## Trello log

### Initial trello board

The screenshot shows a Trello board titled "13DS Assessment". The board has three columns: "Things To Do", "Doing", and "Done".

- Things To Do:**
  - Learn the HCP system in bridge
  - Code setup, (Module import)
  - Read in BRI files
  - do basic testing after import of files is complete and list conversions
  - create lists with all hands (North, East, South,)
  - Calculate fourth hand (West) out of given info of (North, South, East)
  - Calculate the HCP
  - Create a basic GUI with graphical events (key/ button press)
  - Create a question based system
- Doing:** No cards in this column.
- Done:** No cards in this column.

13th of sept

13DS Assessment

Personal | Private | NG | Invite

Things To Do

- Create a basic GUI with graphical events (key/ button press)
- Create a question based system ("What is the HCP of this...")
- create logo (custom made in photoshop)
- add aesthetically pleasing background
- create card initialisation function
- create all turtles for the card
- create positions for these turtles
- make sure each input screen is "robust" by allowing multiple synonyms for yes to be entered
- create the main turtle screen at the specific resolution of (700, 600)
- add the shapes of all the cards into turtle by strategically reading in 2 characters and adding ".gif" to the end with these characters also being all 52 cards "identifier" names and also logically having the same image name as the card format in the root directory.

Doing

- Calculate the HCP

+ Add another card

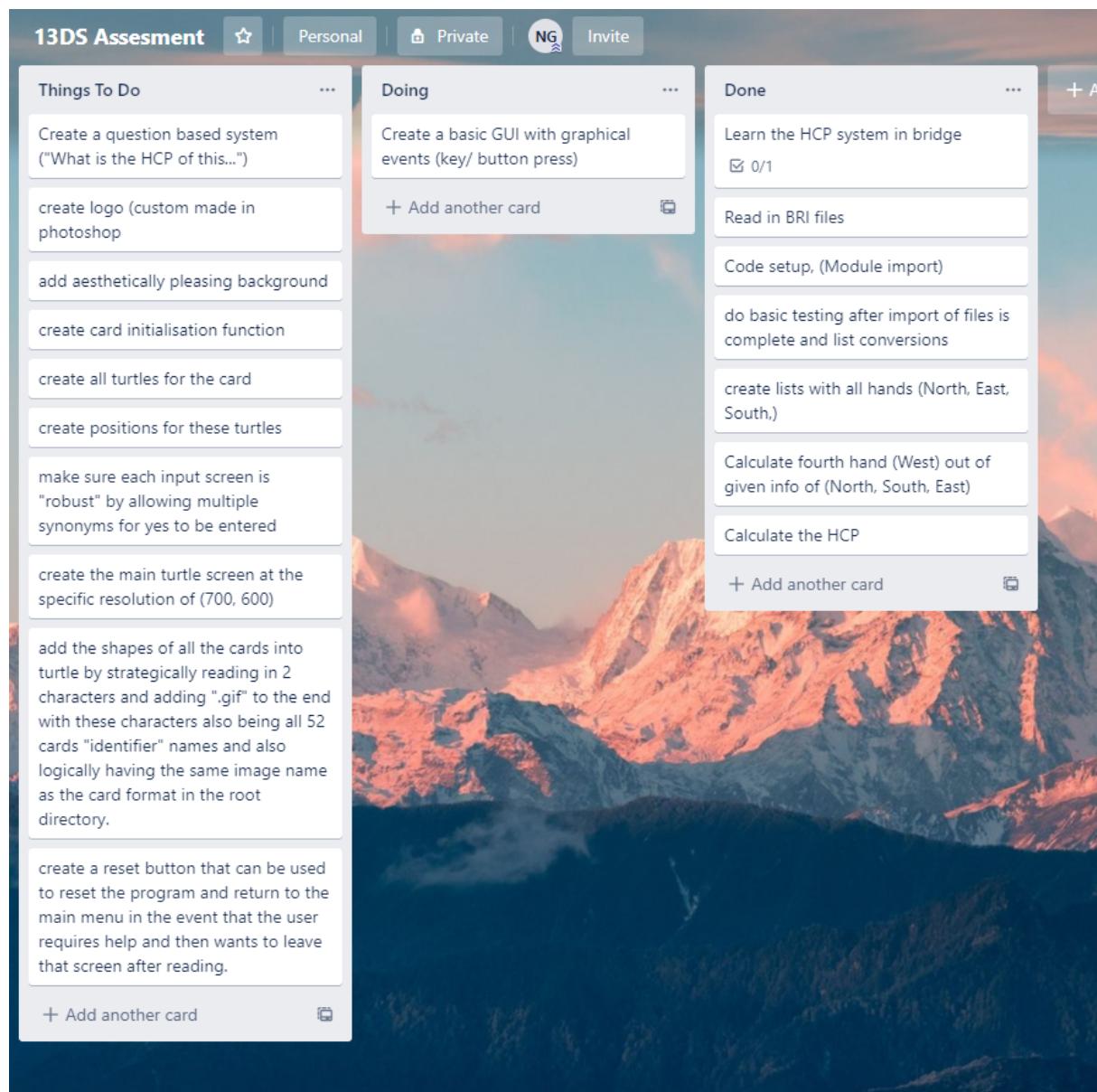
Done

- Learn the HCP system in bridge
- Read in BRI files
- Code setup, (Module import)
- do basic testing after import of files is complete and list conversions
- create lists with all hands (North, East, South,)

+ Add another card

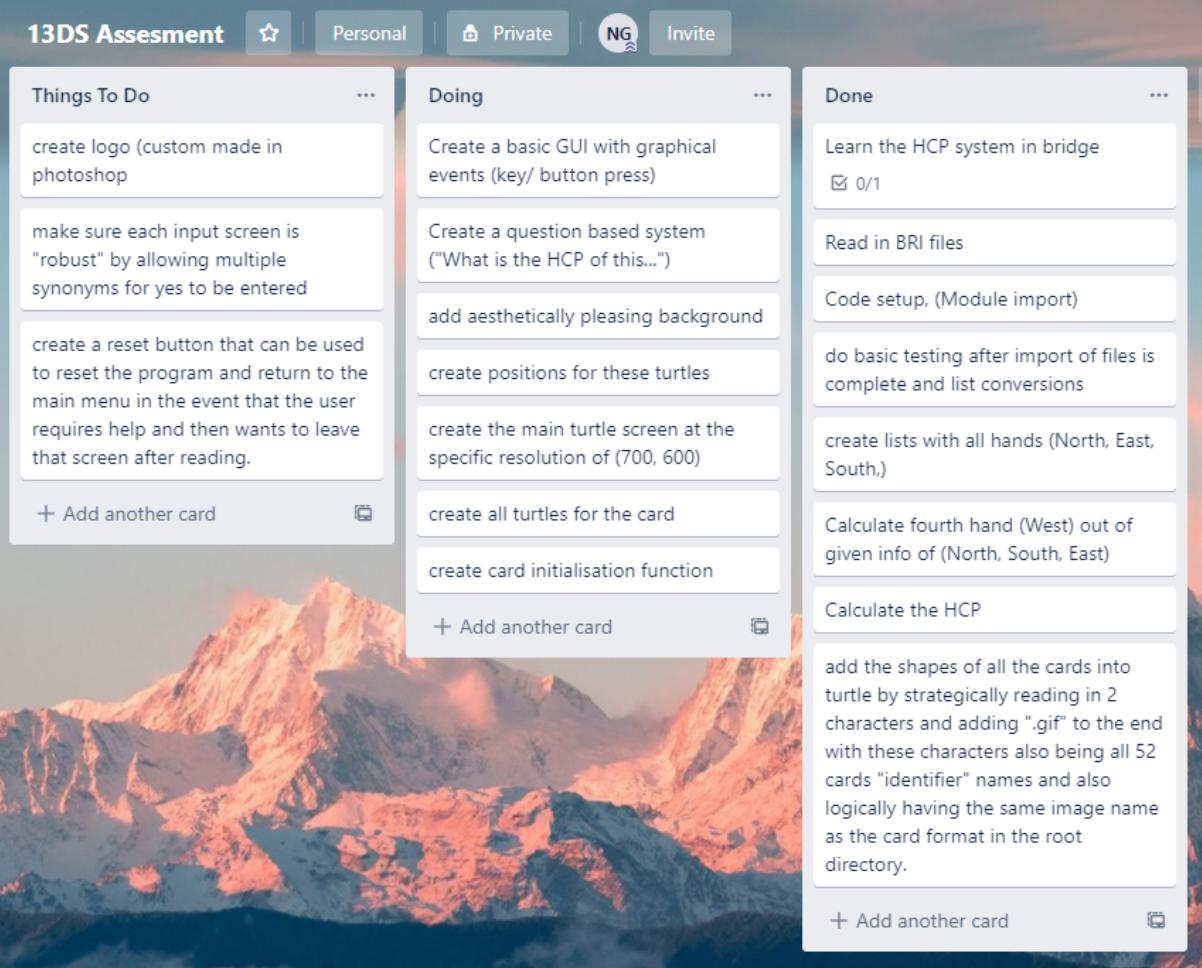
The background of the board is a photograph of a mountain range at sunset, with peaks illuminated in orange and red against a blue sky.

14th



16th

13DS Assessment  Personal  Private  Invite



Things To Do

- create logo (custom made in photoshop)
- make sure each input screen is "robust" by allowing multiple synonyms for yes to be entered
- create a reset button that can be used to reset the program and return to the main menu in the event that the user requires help and then wants to leave that screen after reading.

+ Add another card 

Doing

- Create a basic GUI with graphical events (key/ button press)
- Create a question based system ("What is the HCP of this...")
- add aesthetically pleasing background
- create positions for these turtles
- create the main turtle screen at the specific resolution of (700, 600)
- create all turtles for the card
- create card initialisation function

+ Add another card 

Done

- Learn the HCP system in bridge  0/1
- Read in BRI files
- Code setup, (Module import)
- do basic testing after import of files is complete and list conversions
- create lists with all hands (North, East, South.)
- Calculate fourth hand (West) out of given info of (North, South, East)
- Calculate the HCP
- add the shapes of all the cards into turtle by strategically reading in 2 characters and adding ".gif" to the end with these characters also being all 52 cards "identifier" names and also logically having the same image name as the card format in the root directory.

+ Add another card 

17th

**Things To Do**

- create logo (custom made in photoshop)
- create a reset button that can be used to reset the program and return to the main menu in the event that the user requires help and then wants to leave that screen after reading.

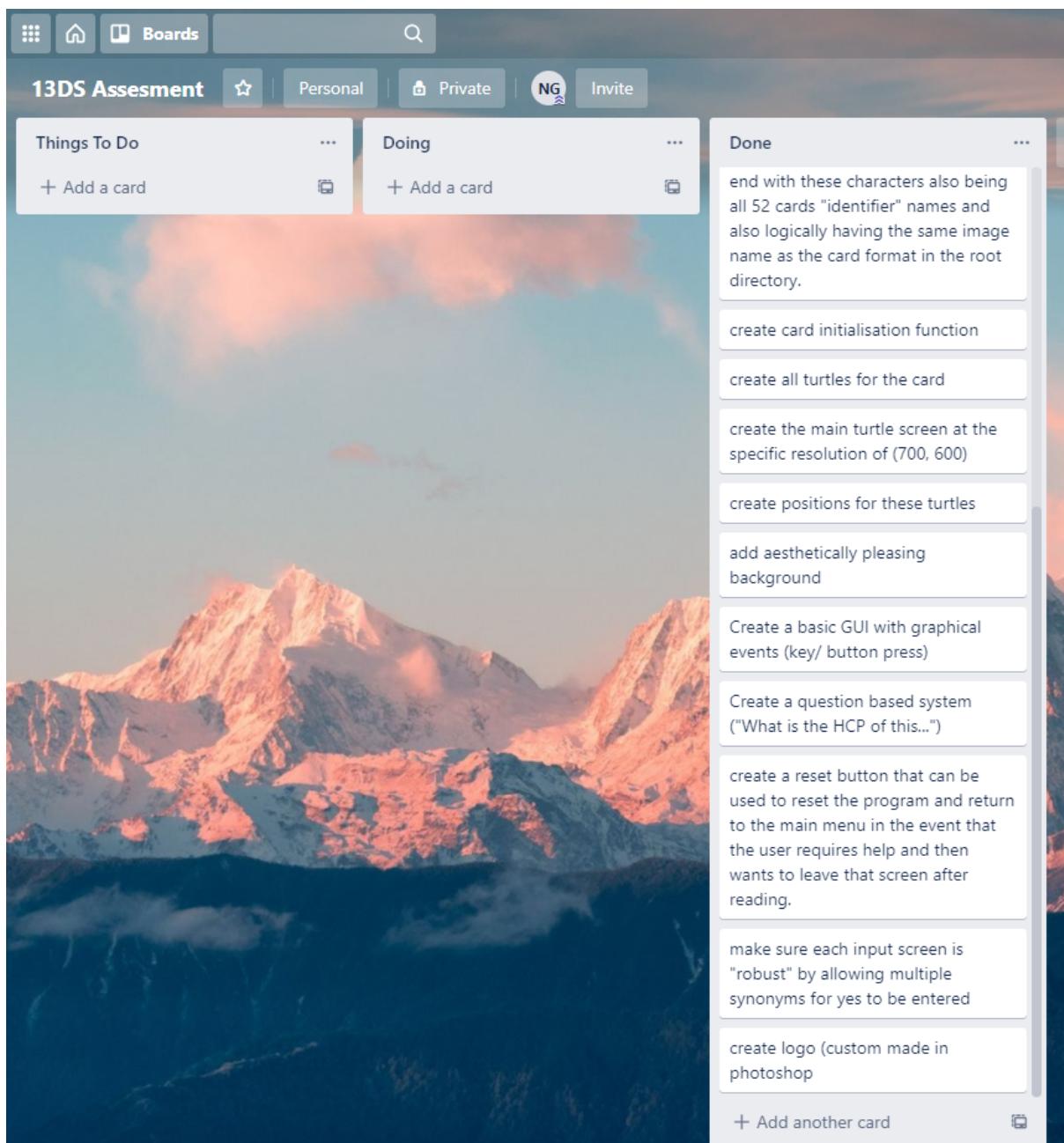
**Doing**

- make sure each input screen is "robust" by allowing multiple synonyms for yes to be entered

**Done**

- Learn the HCP system in bridge  
✓ 0/1
- Read in BRI files
- Code setup, (Module import)
- do basic testing after import of files is complete and list conversions
- create lists with all hands (North, East, South.)
- Calculate fourth hand (West) out of given info of (North, South, East)
- Calculate the HCP
- add the shapes of all the cards into turtle by strategically reading in 2 characters and adding ".gif" to the end with these characters also being all 52 cards "identifier" names and also logically having the same image name as the card format in the root directory.
- create card initialisation function
- create all turtles for the card
- create the main turtle screen at the specific resolution of (700, 600)
- create positions for these turtles
- add aesthetically pleasing

18th



## Relevant Implications:

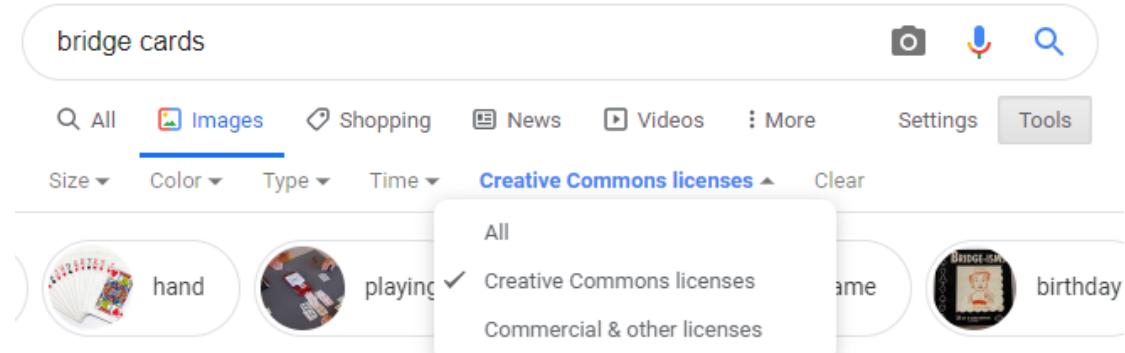
### Aesthetics:

Aesthetics are a huge part of programming and is essential to having a successful program. I also incorporate some “localization” by creating a custom logo incorporating the Paraparaumu College logo, this will be aesthetically pleasing and will be made by me in Adobe Photoshop.

Image ethics, when it comes to using images in a program one of the most important things is following copyright laws and ensuring that the image you have used is legally allowed to be used, in New Zealand the copyright law is that it gives

the person who creates an original work exclusive rights to copy, publish, publicly perform, transmit and adapt their material. The basic premise of copyright law is that the creator of the works (in most cases, the author is the person who creates a work) has the right to decide how their work will be used. This is why it's important to see the legalities of using an image before going onto a search engine and using whatever image you like.

What i have done to ensure that there are no copyright issues is that i have selected the creative commons license filter on google images:



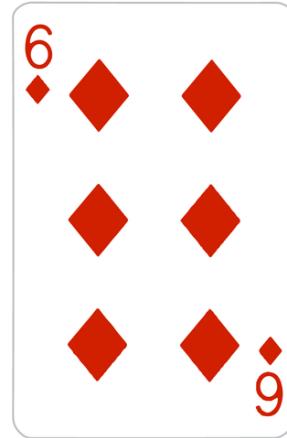
Although this doesn't necessarily ensure that all images are copyright free so to ensure that i have proceeded to go onto the website and check the copyright information as seen in this picture:



This was the same with the reset button image I used along with the college logo image as that is allowed due to this project being for the college club and the image falling under fair use especially as the program is for the college.

The images for the actual cards were sourced from the American contract Bridge League website and were a downloadable zip that is allowed to be reused commercially.

Any other images were created by myself through photoshop and are allowed to be used. What this means is that the program is fully legal in terms of image use.



#### Future proofing:

To ensure that the program is future proofed, I have made use of a popular programming language that isn't going anywhere in the near future, the language i have used is Python, I have also made sure that my program is fully compatible with the latest version of python and this is to ensure compatibility for at least the next big iterations of python which adds another layer to the future proofing of the program. Another aspect of future proofing is using comments, this is so that yourself or someone else that may be using the program in the future will find it very easy to use due to the program having easy to understand comments therefore being easy to recode sections that are needed due to differences in new python versions.

#### Sustainability:

The sustainability of the program can be maintained by ensuring that everything is local within the program, this essentially means that the program doesn't need to connect to external servers for any authentication or for any files and is instead optimized and compressed enough so that it can be stored locally and be easily accessible anywhere, anytime.

#### Readability:

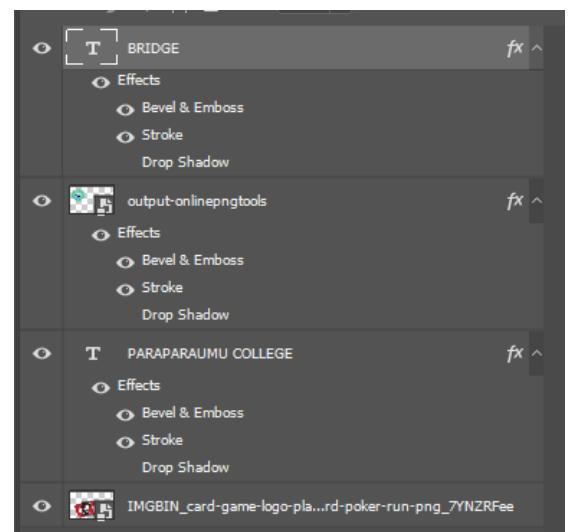
The readability of the program can be defined as how easy it is to read the program, it is for this reason that I have made sure that all the variable names are "logical" and make sense along with calling the functions relative to what the function actually does, this uniformity ensures that the program is easy to read if another person was going through it and were attempting to make changes.

## Advanced Techniques:

The images used for the cards were downloaded as a batch of 52 cards in the png format, they were then saved once through photoshop while using an automation script recorder which detected the changes made on the first card manually, i then used the batch script option in photoshop to feed in the folder of images that weren't manipulated and set the output directory to another folder and ran it. This

made all the images in the folder convert to a “.gif” format while retaining the transparent background, I also resized the images through this script to ensure that the images were the perfect size to be read easily as many images in their stock configuration were too small making it difficult to read.

The logo was created in photoshop by grabbing an image of the cards from the website “imgbin” and then proceeding to manipulate the image by adding the PC logo along with the tech “Paraparaumu college bridge” with the effects (bevel and emboss, outline and dropshadow applied) i used these effects on the text along with the PC logo to add depth and to make it match with the original cards image, in the end the colour scheme is pleasing to the eye and so is the design.



## User feedback and testing:

For user testing a got patrick howe to go and test my program, doing this allowed me to spot some critical bugs and errors in my program, one of them was that if the user when asked what hand to draw inputted an invalid response they would then proceed to have the cards drawn with the turtles but with no images and with the program hanging at that point, to fix this i created an “if, else” statement and made sure that the program always either gave an error explaining why the input was wrong and what they could do to fix this, if the user has an input error again then the program would loop back to the main menu, what this did is ensure that the program never hanged or got stuck for the user.

Another bug encountered was that the function for grabbing an item from a list wasn't executing correctly, after further investigation this was due to the use of index numbers in python where you start counting from zero, so when my code said “choose first item from the list

```
def northhand_cardgen():
    #this function generates the cards by pulling out values out of a list that is created
    #by comparing against a database
    global cardpics_raw  #making the variable accessible outside the function
    cardpics_raw = [
        card_database[k] for k in northhand
    ] #generates list for the hand showing all the card identifier names in the hand.

    #adds ".gif" to the end of every item in the list to select the images that are being
    #displayed
```

and add “.gif” to it instead of the first item it ended up choosing the second item, to fix this i had to shift all the numbers by 1 and start counting from 0 all the way to 12 as seen in this image:

Another issue i had was that when the user would complete the program (for ex, they choose north hand and got the hcp correct and wanted to calculate the HCP of another hand) the issue here was that the old card turtles would remain and stay there until the new hand was selected, this definitely made some confusion occur when discussing with my peers and helped me make the informed decision to investigate a better way of continuing. To resolve this issue what i did was that whenever the “main()” function was called the program would make sure that it resets the screen so essentially every turtle that was displayed on screen would be reset to its original state and ready to be manipulated again, once this was done it made the program feel more “polished” and professional as it was much easier to use.

Another issue i encountered was that the resolution of the card images provided on the colleges website “PC4ME” were a bit small and this was due to the resolution being lowered and without being able to easily make size changes in the limited python turtle i made some changes, firstly i went to the american contract bridge association where they had a 52 pack of cards up for download and allowed for commercial use and then went ahead and created a photoshop batch task that converted each image to the size of 83 pixels x 127 pixels and then converting all the files to a “.gif” with a photoshop automation script job which was the perfect dimensions to have the cards displaying at the correct size allowing for easy readability for the student playing the program.

## Interoperability - Different BRI Files

One of the main purposes of programs is for them to be very interoperable and for them not to be fixated, what this means is that the program is more flexible and robust and allows the end users (bridge club) to try many, many different variations of the BRI files. I tested a few BRI files from the [NZ Bridge](#) website and tried many different BRI files, as my program was designed with interoperability in mind to ensure that the program is more robust and a better overall program that it would have been without this in the making.



### Expected cases program testing

Question asked	Input	Expected behaviour	Actual behaviour
What hand do you want to calculate the HCP of?	North	Loads north hand and asks what HCP they want to enter	Loads north hand and asks what HCP they want to enter
What is the HCP of the displayed north hand?	14	14 is indeed the correct answer and the program should congratulate them and have another prompt	14 is indeed the correct answer and the program should congratulate them and have another prompt
Would you like to return to the start	yes	Loads (main()) which should bring back to	Loads (main()) which should bring

of the program?		the main menu	back to the main menu
What hand do you want to calculate the HCP of?	east	Loads easthand and asks what HCP they want to enter	Loads easthand and asks what HCP they want to enter
What is the HCP of the displayed east hand?	7	Should say that the answer entered was incorrect and should ask the user if they want help	Should say that the answer entered was incorrect and should ask the user if they want help
The answer you entered for the HCP was incorrect. Would you like some help?	yes	Should load up the definition of the HCP, along with what each cards point value is, it should also say what the correct HCP was so that the user knows what the correct HCP was, now it should also display a reset button	Should load up the definition of the HCP, along with what each cards point value is, it should also say what the correct HCP was so that the user knows what the correct HCP was, now it should also display a reset button
Reset button displayed on the bottom of the screen	Clicking the reset button	This should load the ("main()") function and should in theory restart the program, when this happens the user will return to the start of the program.	This should load the ("main()") function and should in theory restart the program, when this happens the user will return to the start of the program.
What hand do you want to calculate the HCP of?	east	Loads east hand and asks what HCP they want to enter	Loads east hand and asks what HCP they want to enter

## Unexpected cases program testing

### Boundary cases program testing

This is testing boundary cases within the program and evaluating what the response from the program is and iterating on this to ensure the program is better.

Question asked	Input	Expected behaviour	Actual behaviour
What hand do you want to calculate the HCP of?	any	To reject the input as an invalid input and then proceed to give a retry prompt along with instructions of what inputs are accepted	To reject the input as an invalid input and then proceed to give a retry prompt along with instructions of what inputs are accepted
You entered and invalid input, please try again:	Not sure	To run the main() function which resets and runs the program from the start, this should take the user back to the main “What hand do you want to calculate the HCP of?” prompt.	To run the main() function which resets and runs the program from the start, this should take the user back to the main “What hand do you want to calculate the HCP of?” prompt.
What hand do you want to calculate the HCP of?	south	Loads south hand and asks what HCP they want to enter	Loads south hand and asks what HCP they want to enter
What is the HCP of the displayed south hand	eight	As this is a text input within an integer input python spits out an “illegal value, not a floating point value, please try again” error which lets the user know that they should enter an integer instead.	As this is a text input within an integer input python spits out an “illegal value, not a floating point value, please try again” error which lets the user know that they should enter an integer instead.
What is the HCP	nine	As this is a text	As this is a text

of the displayed south hand		input within an integer input python spits out an "illegal value, not a floating point value, please try again" error which lets the user know that they should enter an integer instead.	input within an integer input python spits out an "illegal value, not a floating point value, please try again" error which lets the user know that they should enter an integer instead.
What is the HCP of the displayed south hand	-1	Should say that the HCP entered is incorrect, would you like some help?	Should say that the HCP entered is incorrect, would you like some help?
The answer you entered for the HCP was incorrect. Would you like some help?	yes	Should load up the definition of the HCP, along with what each cards point value is, it should also say what the correct HCP was so that the user knows what the correct HCP was, now it should also display a reset button	Should load up the definition of the HCP, along with what each cards point value is, it should also say what the correct HCP was so that the user knows what the correct HCP was, now it should also display a reset button
Reset button displayed on the bottom of the screen	Clicking the reset button	This should load the ("main()") function and should in theory restart the program, when this happens the user will return to the start of the program.	This should load the ("main()") function and should in theory restart the program, when this happens the user will return to the start of the program.
What hand do you want to calculate the HCP of?	south	Loads north hand and asks what HCP they want to	Loads north hand and asks what HCP they want to

		enter	enter
What is the HCP of the displayed north hand?	38	Should load up the definition of the HCP, along with what each cards point value is, it should also say what the correct HCP was so that the user knows what the correct HCP was, now it should also display a reset button	Should load up the definition of the HCP, along with what each cards point value is, it should also say what the correct HCP was so that the user knows what the correct HCP was, now it should also display a reset button

Here two of the inputs (-1), (38) are both “boundary cases” with the smallest HCP possible being 0, and the largest possible HCP being 37, what this means is that the values entered previously (-1) and (38) should not be valid as they are impossible. For good programming practise and to lead the player on the right path along with making sure that the user did not make a typo and accidentally entered an extra digit what should be done is that any numbers above or below the ones specified should be rejected.

### Boundary cases after being fixed

Question asked	Input	Expected behaviour	Actual behaviour
What hand do you want to calculate the HCP of?	North	Loads north hand and asks what HCP they want to enter	Loads north hand and asks what HCP they want to enter
What is the HCP of the displayed north hand?	-1	The program has now been modified to handle invalid inputs, should reject the input and tells the user that the HCP can't be smaller than 0 or greater than 37 (0 is smallest and	The program has now been modified to handle invalid inputs, should reject the input and tells the user that the HCP can't be smaller than 0 or greater than 37 (0 is smallest and

		37 is largest)	37 is largest)
The HCP can't be smaller than 0 or greater than 37! Type yes to try again or no to return to the main menu	yes	Runs the hcp function allowing the user to try again and re-enter the HCP	Runs the hcp function allowing the user to try again and re-enter the HCP
What is the HCP of the displayed north hand?	38	The program has now been modified to handle invalid inputs, should reject the input and tells the user that the HCP can't be smaller than 0 or greater than 37 (0 is smallest and 37 is largest)	The program has now been modified to handle invalid inputs, should reject the input and tells the user that the HCP can't be smaller than 0 or greater than 37 (0 is smallest and 37 is largest)
The HCP can't be smaller than 0 or greater than 37! Type yes to try again or no to return to the main menu	yes	Runs the hcp function allowing the user to try again and re-enter the HCP	Runs the hcp function allowing the user to try again and re-enter the HCP
What is the HCP of the displayed north hand?	14	As 14 is a valid input and is also the correct HCP the program should allow the user to return to the start of the program or exit if they wish.	As 14 is a valid input and is also the correct HCP the program should allow the user to return to the start of the program or exit if they wish.
Would you like to return to the start of the program?	yes	Runs the main() function which therefore re-runs the program and returns to the start.	Runs the main() function which therefore re-runs the program and returns to the start.

## Unexpected case program testing

Question asked	Input	Expected behaviour	Actual behaviour
What hand do you want to calculate the HCP of?	random	Should show a prompt showing "invalid input" and should allow the user to enter again	Should show a prompt showing "invalid input" and should allow the user to enter again
You entered an invalid input, please try again	test	Rejects the input and runs the main() function again which essentially restarts the program.	Rejects the input and runs the main() function again which essentially restarts the program.
What hand do you want to calculate the HCP of?	testing	Should show a prompt showing "invalid input" and should allow the user to enter again	Should show a prompt showing "invalid input" and should allow the user to enter again
You entered an invalid input, please try again	North	Loads north hand and asks what HCP they want to enter	Loads north hand and asks what HCP they want to enter
What is the HCP of the displayed north hand?	random	As this is a text input within an integer input python spits out an "illegal value, not a floating point value, please try again" error which lets the user know that they should enter an integer instead.	As this is a text input within an integer input python spits out an "illegal value, not a floating point value, please try again" error which lets the user know that they should enter an integer instead.
What is the HCP of the displayed north hand?	-50	Rejects input and shows prompt explaining that	Rejects input and shows prompt explaining that

		the smallest hcp can be 0 and the largest can be 37 (these are the smallest and largest possible HCP values)	the smallest hcp can be 0 and the largest can be 37 (these are the smallest and largest possible HCP values)
The HCP can't be smaller than 0 or greater than 37! Type yes to try again or no to return to the main menu	y	Runs the hcp function allowing the user to try again and re-enter the HCP	Runs the hcp function allowing the user to try again and re-enter the HCP

What this shows is that the program is able to fully cope with “unexpected” values and rejects them while giving proper explanations, overall this error checking improves the quality of life for the program and allows the user to understand properly

This video below is a basic run through of the program

[Video demonstrating the program and testing boundary cases](#)

### Summary:

To summarise many things could have been changed in the development of the program, I could have made a few differences to improve the program and reach an evaluation that was better for my stakeholders.

One thing that could have been improved is time management, with tools such as trello available I believe that I potentially could improve how I managed time, especially as the time provided was fairly low when compared to what was needed. I believe that the trello management could have been in more detail which would in turn mean that my stakeholders would have better communication with me while I was developing the program. Overall the program ended up working well and did what it's purpose was without any issues whatsoever.