# Independent Study Project – Checkpoint 1

## Purpose

To create a product that engages you and that you would be proud to share to a public audience.

Along the way, you will develop your ability to problem-solve using a variety of strategies, to implement a solution in code, to manage source code using accepted industry practices, and to plan and meet commitments for project milestones.

## Evaluation

As described in January, I am now taking a standards-based approach to evaluating your progress in the course.

What does that mean?

It means that I value the *process* of your work on this ISP as much as your *product.*

It means that I am looking, quite simply, for you to provide evidence of having met the expectations listed.

To that end: using your commits on GitHub, and your posts on Sesame, how would *you* evaluate your progress so far?

You probably will not have yet demonstrated *all* of the expectations, but have you hit some? How often?

For each expectation shown on the following pages:

1. Provide links(s), optionally with brief explanatory text to specific parts of a commit in your source control history
2. Give yourself a 1 to 5 star rating

## Curriculum Expectations

### A1. Data Types and Expressions Demonstrate the ability to use different data types, including one-dimensional arrays, in computer programs;

**A1.1** use constants and variables, including integers, floating points, strings, and Boolean values, correctly in computer programs;

ASCII, Unicode) to internally represent data and store information;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/commit/0e1cc005edff79c7ce5b3de97b30a6094ba947d8#diff-774ded294570492e07856a032c2127c5R31>  This is one of the structures used in my program, all the information is stored in different variables |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L66-L84>  This code sets up the URL session to get my JSON data. It uses constants for everything except for the urlRequest variable because this is the only value that is altered later in the code. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/ef82ccdb19acaf8140cc57d4cbea60a4268d161a/Football%20Match%20Predictor/ViewController.swift#L124-L130>  This stores all of the JSON data that I pull for the teamData. It stores them as constants because the values are not ever changed. They are held as constants and then a temporary team class is constructed from the data. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A1.3** use assignment statements correctly with both arithmetic and string expressions in computer programs;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| https://github.com/rsgc-logush-o/Football-Match-Predictor/commit/0e1cc005edff79c7ce5b3de97b30a6094ba947d8#diff-c4701aa841915a28932b648e506afc6aR47  This is an example in my code of using assignment statements to set a value to a variable |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L132-L140>  This is an example of assignment statements being used correctly. I am using let statements here to create constants from the data I obtained from the JSON. I am using constants because the values are not being altered after the assignment. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A1.4** demonstrate the ability to use Boolean operators (e.g., AND, OR, NOT), comparison operators (i.e., equal to, not equal to, greater than, less than, greater than or equal to, less than or equal to), arithmetic operators (e.g., addition, subtraction, multiplication, division, exponentiation, parentheses), and order of operations correctly in computer programs;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/374bd46fb42b3fba5509c49ef10a59c82bdec7a9/Football%20Match%20Predictor/ViewController.swift#L91-L102>  This function uses both addition, subtraction, and multiplication to obtain an average that I needed. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/ef82ccdb19acaf8140cc57d4cbea60a4268d161a/Football%20Match%20Predictor/ViewController.swift#L191>  This checks the value of the status code that the URLRequest gives us. If the code is 200 which means that the request is OK, we continue with the remaining code, if not print(“Error cannot create URL object”) |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A1.5** describe the structure of one-dimensional arrays and related concepts, including elements, indexes, and bounds;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/commit/0e1cc005edff79c7ce5b3de97b30a6094ba947d8#diff-774ded294570492e07856a032c2127c5R26>  This is an array of a struct that I am using to store the matches. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L57>  This array holds all of the matches played throughout the season. It is declared with no elements in it, however, later in the program the match data is stored it it. The datatype that is stored is the “Match” structure. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/ef82ccdb19acaf8140cc57d4cbea60a4268d161a/Football%20Match%20Predictor/ViewController.swift#L138-L141>  This is when |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A1.6** write programs that declare, initialize, modify, and access one-dimensional arrays.

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/commit/0e1cc005edff79c7ce5b3de97b30a6094ba947d8#diff-c4701aa841915a28932b648e506afc6aR91>  This goes through the matches played which is stored in an array |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L123-L126>  This initializes an array of any object meaning that the type is not specified for the data. This is because JSON provides many different types of data. It then loops through each object in the array and seperates it and then eventually parses it and assigns the values. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/ef82ccdb19acaf8140cc57d4cbea60a4268d161a/Football%20Match%20Predictor/ViewController.swift#L138-L141>  This is when I am storing data to the dictionary of teams. I am first making a structure to temporarily hold the data that I will then store to the dictionary. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

### A2. Control Structures and Simple Algorithms Demonstrate the ability to use control structures and simple algorithms in computer programs;

**A2.1** write programs that incorporate user input, processing, and screen output;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/c9db7d4cc2e50d1575e711fea354b88dbc383f7f/Football%20Match%20Predictor/ViewController.swift#L316-L331>  This function takes the two teams that the user wants to match against each other. I had difficulty with my user interface so the user needs to enter the names in code. It does take their input, process the data and compare it, then output the result. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/c9db7d4cc2e50d1575e711fea354b88dbc383f7f/Football%20Match%20Predictor/ViewController.swift#L277-L313>  This is where most of the processing is done for the program. It is done before the user input because all of the team’s stats are already determined when the data is collected. This function is what determines the most important statistc for each team, the average margin that the team outperforms their opponents averages. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A2.2** use sequence, selection, and repetition control structures to create programming solutions;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/374bd46fb42b3fba5509c49ef10a59c82bdec7a9/Football%20Match%20Predictor/ViewController.swift#L91-L102>  This function uses a loop to go through all of the data that needs to be averaged. It repeats addition and subtraction until all of the needed elements have been used then it divides the result to obtain the mean. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L126-L163>  This repeats through all of the data sets in the JSON to assign the data to a team. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A2.3** write algorithms with nested structures (e.g., to count elements in an array, calculate a total, find highest or lowest value, or perform a linear search).

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/c9db7d4cc2e50d1575e711fea354b88dbc383f7f/Football%20Match%20Predictor/ViewController.swift#L277-L313>  This is an algorithm that finds average scoring and defense margins, it is the most important function in my code. It goes through all of the teams, then finds the matches they are in and determines how much they outperformed/underperformed their opponents average defense/offense. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L126-L164>  This goes through all of the JSON data and assigns the values to the needed teams in the program.  <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/c9db7d4cc2e50d1575e711fea354b88dbc383f7f/Football%20Match%20Predictor/ViewController.swift#L280-L310>  As requested this part takes goes through all of the matches and finds the data relating to the team that needs to be obtained. Then finds the average margins. This is the piece of code that I am the most proud of because it is unique and orginially was a difficult problem to tackle because the data was not presented in a fashion that fit my data structure well. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

### A3. Subprograms Demonstrate the ability to use subprograms within computer programs;

**A3.1** demonstrate the ability to use existing sub-programs (e.g., random number generator, substring, absolute value) within computer programs;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L69-L84>  This uses a URL sub program to obtain my JSON data needed for the program. I use many aspects of the sub program to get the data. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L230>  This subprogram serializes the JSON so that I can later parse it. It is needed in my program. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/c9db7d4cc2e50d1575e711fea354b88dbc383f7f/Football%20Match%20Predictor/ViewController.swift#L177>  This subprogram is a part of the URL class. It adds the API key value to the instance of the URL class I am using. I needed this because the API I was accessing needed a key to access it. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A3.2** write subprograms (e.g., functions, procedures) that use parameter passing and appropriate variable scope (e.g., local, global), to perform tasks within programs.

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L118-L171>  This function gets JSON data from a source that I will later parse. It then sends the information into a new subprogram that does the parsing. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8b50cfc1be545905899aa93f75f49ca5a665c454/Football%20Match%20Predictor/ViewController.swift#L105-L116>  This passes the team that needs the calculations to be done on. It uses the information inside of this teams structure. It then returns the value that was obtained. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/c9db7d4cc2e50d1575e711fea354b88dbc383f7f/Football%20Match%20Predictor/ViewController.swift#L277-L313>  This function that I wrote iterates through all of the matches in the season and finds which ones the team of interest has played in, then obtains the other teams averages, and then compares the scores to see how the over/underperformed their averages. This is the largest chunk of statistical analysis code. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

### A4. Code Maintenance Use proper code maintenance techniques and conventions when creating computer programs.

**A4.1** demonstrate the ability to identify and correct syntax, logic, and run-time errors in computer programs;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| When initializing an instance of the team structure in my code to temporarily hold JSON data that I had collected I ran into a syntax error. This turned out to be because I did not format the initialization correctly.  <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/04db295798c299ee53009749fc7757049adafbce/Football%20Match%20Predictor/ViewController.swift#L129>  This is the line of code that was not working after I had fixed it, here is a picture of before and after.  ../../Desktop/Screen%20Shot%202017-05-18%20at%201.38.49%20PM.pn    ../../Desktop/Screen%20Shot%202017-05-18%20at%201.36.31%20PM.pn |
|  |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A4.2** use workplace and professional conventions (e.g., naming, indenting, commenting) correctly to write programs and internal documentation;   
 (also includes use of source control)

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L8-L272>  This is an example of my code being well documented and using proper naming conventions with camel casing. |
|  |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A4.3** demonstrate the ability to interpret error messages displayed by programming tools (e.g., compiler, debugging tool), at different times during the software development process (e.g., writing, compilation, testing);

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| ../../Desktop/Screen%20Shot%202017-04-24%20at%208.53.01%20PM.pn  Here I was using a let statement to define urlRequest because before I did not need to edit the value. This is because I needed to add an api key to the request. It gave me an error because it since I needed to edit it I needed to declare it as a variable not a constant. I did that and the problem was solved.  ../../Desktop/Screen%20Shot%202017-04-24%20at%208.53.30%20PM.pn |
|  |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A4.4** use a tracing technique to understand program flow and to identify and correct logic and run-time errors in computer programs;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/04db295798c299ee53009749fc7757049adafbce/Football%20Match%20Predictor/Info.plist>  This is where I needed to add a permission to get my api access working. The issue was my permissions were not allowing access to the link. I fixed this by finding the error my code provided using the swift debugging tool. The link provided is the permissions page that I needed to edit. |
|  |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**A4.5** demonstrate the ability to validate a program using a full range of test cases.

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| My program is able to collect all of the data however, it tries to run the team comparing function before the data collection functions have completed which I was unable to fix. |
|  |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

### B1. Problem-solving Strategies Use a variety of problem-solving strategies to solve different types of problems independently and as part of a team;

**B1.1** use various problem-solving strategies (e.g., stepwise refinement, divide and conquer, working backwards, examples, extreme cases, tables and charts, trial and error) when solving different types of problems;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/04db295798c299ee53009749fc7757049adafbce/Football%20Match%20Predictor/ViewController.swift#L113-L121>  <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L132-L140>  Here I used the debugging tool in swift to find my error when breaking apart the data and storing it. The issue was I had used the wrong term to access data from the dictionary. |
|  |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

**B1.2** demonstrate the ability to solve problems independently and as part of a team;

| Evidence: provide link(s) where possible, optionally provide brief explanatory text, add rows as needed |
| --- |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/e94301698ddc5b5526d294cd628c554025965a05/Football%20Match%20Predictor/Base.lproj/Main.storyboard>  An example of solving problems as a part of a team is when working on my UI. Although my UI was not completed in the end, I got help with using the viewController tools from Ethan Peterson, Ethan McCaulliffe, and Spencer Canavan. Before talking to them I had a lot of difficulty getting my UI working and after talking to them I had something that displayed on the screen. |
| <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/8f2c9c9fe3ea622ae2e5efc8835434ed51ee64da/Football%20Match%20Predictor/JSON.playground/Contents.swift#L63-L115>  <https://github.com/rsgc-logush-o/Football-Match-Predictor/blob/04db295798c299ee53009749fc7757049adafbce/Football%20Match%20Predictor/Info.plist>  An example of working in a team is when I was having difficulty accessing the API. We looked over the code together, and that was fine, however he helped figure out that there were specific permissions that I needed to enable. |

**Overall rating on this standard**: ✩ ✩ ✩ ✩ ✩

## Comments and Proposal for Level of Achievement

Understanding that this is a checkpoint 1/3 of the way into the ISP, and that mastery of all standards is not expected at this point in time, what do you suggest as your current level of achievement? Why?