**PLANNING A COMPLEX ALGORITHM**

**DESIGN THE ROUTINE**

CHECK PREREQUISITES

Define the problem

*Our Code has redundant functions, is not lean, has some syntax and data generation problems, and does not display correctly on our page*

Information the routine will hide

*All previously calculated routines will remain the same and work in the background*

Inputs to the routine

*-*

Outputs from the routine

*Displays Text in a more aesthetically pleasing way in Separate Divisions*

Pre-conditions

*All Data is calculated and inputted. The Information is already set up into their appropriate divs, and all DIVs are targetable by either class or id*

Post-conditions

*Data is now displayed more prettily on the screen from dynamic generation with html dom elements, also all text is pretty and displays properly. No values are missing from the data generation*

Name the Routine

*Fixing result display from The Commonwealth Games 2018*

Decide how to test the routine

*All data should display correctly, without the use of some place holder data. We should also see some column implementation and styling applied to stop excess spewing of irrelevant styling data.*

Research functionality available in standard libraries

*The majority of this assignment is working with HTML DOM, which is readily available within JS Libraries. I will have to Learn how to assign attributes to divs and learn some css to be able to apply it.*

Think about error handling

*Generally, with this dynamic generation Physical errors are able to be identified and fixed as data will not return on the page*

Think about efficiency

*Looping will be used to generate the majority of the data. This approach will avoid the Hard Coding of every single result.*

Research algorithms & data types

*Dom Elements have a different syntax of assigning variables and attributes than regular html. Assigning them to a variable then using functions to assign these attributes seems to be the simplest way of doing it.*

**WRITE PSEUDOCODE**

1. Think about the data
2. Check the pseudocode
3. Try ideas in pseudocode

**CODE THE ROUTINE**

1. Write the declaration
2. Turn pseudocode into comments
3. Fill in code below comments
4. Check if code can be factored

**CHECK THE CODE**

1. Mentally check for errors
2. Step through in Debugger
3. Test the code
4. Remove errors in the code
5. Clean up