## Question Fifteen

### Instructions

Fill in all sections of the Test Report, start by completing the Project Details. Then list all the Test Cases and the associated Test Steps. Run the tests and record the results. Review the results and modify the code to ensure correct functionality of the application.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Report | | | | | | | | |
| Project Details: Astronomical Processing Program | | | | | | | | |
| Date | | 20/10/21 | | | | | | |
| Team Name | |  | | | | | | |
| Team Members | | Shaun Rawlings | | | Scott King | | | |
| Test Type | | Testing new buttons added in version 4 of program | | | | | | |
| Test Case # | Test Case Name | | Test Steps | Test Data | | Expected Results | Evidence  (ref to Screen Capture) | Pass / Fail |
| 1 | Average test | | **I**nsert breakpoint on line 335 to test average for loop.  Run and add watch points to track Int i, sum of array, array of numbers, and text box output | **Nothing entered into array and look for average** | | NaN | Ref Fig 1 | Pass |
| 2 | Average test | | **I**nsert breakpoint on line 335 to test average for loop.  Run and add watch points to track Int i, sum of array, array of numbers, and text box output | Array filled with numbers 10 through to 33 and search for average | | 21.5 | Ref Fig 2 | Pass |
| 3 | Average test | | **I**nsert breakpoint on line 335 to test average for loop.  Run and add watch points to track Int i, sum of array, array of numbers, and text box output | Array filled with random numbers and search for average | | 45.75 | Ref fig 3 | Pass |
| 4 | Average test | | **I**nsert breakpoint on line 335 to test average for loop.  Run and add watch points to track Int i, sum of array, array of numbers, and text box output | Array filled with the same number “10” | | 10 | Fig 4 | Pass |
| 5 | Mid extreme test | | Insert breakpoint line 302 to test max value loop.  Add watch points to track Int I and max value return. | Array filled with numbers 10 through to 33 and search for max value | | 33 | Fig 5 | Pass |
| 6 | Mid extreme test | | Insert breakpoint line 316 to test min value loop.  Add watch points to track Int I and min value return. | Array filled with numbers 10 through to 33 and search for min value | | 10 | Fig 6 | Pass |
| 7 | Mid extreme test | | Insert breakpoint line 329 to test mid extreme output. Watchpoint at textboxoutput | Array filled with numbers 10 through to 33 and search for mid extreme value | | 21 | Fig 7 | Fail  Formula was max-min instead of add |
| 8 | Mid extreme test | | Insert breakpoint line 329 to test mid extreme output. Watchpoint at textboxoutput | Array filled with numbers 10 through to 33 and search for Range | | 21 | Fig 8 | Pass |
| 9 | Range test | | Insert breakpoint line 292 to test mid extreme output. Watchpoint at textboxoutput | Array filled with numbers 10 through to 33 and search for Range | | 23 | Fig 9 | Pass |
| 10 | Range test | | Insert breakpoint line 292 to test mid extreme output. Watchpoint at textboxoutput | Array filled with random numbers and search for Range | | 86 | Fig 10 | Pass |
| 11 | Mode Test | | Insert breakpoint line 289 to test Mode output. Add watchpoints to track int frequency and int mode.  mode = ModeOutput | Array filled with random numbers and searched through to find mode | | 43 | Fig 10  Mode variable = 43 | Pass |
| 12 | Mode Test | | Insert breakpoint line 289 to test Mode output. Add watchpoints to track int frequency and int mode.  mode = ModeOutput | Array filled with random numbers and searched through to find mode | | 36 | Fig 11  Mode variable = 36 | Pass |
| 13 | Mode Test | | Insert breakpoint line 289 to test Mode output. Add watchpoints to track int frequency and int mode. int mode = ModeOutput | Array filled with random numbers and searched through to find mode | | 69 | Fig 12  Mode variable = 69 | Pass |

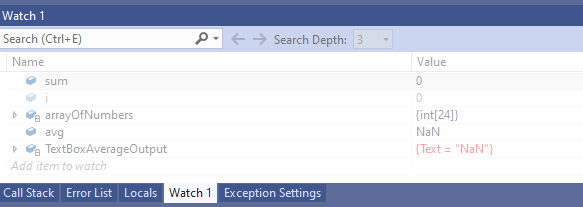


Fig 1

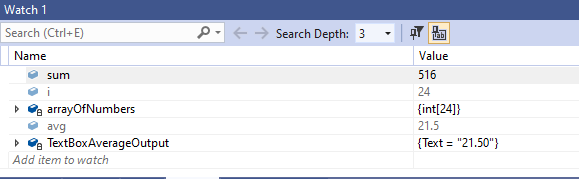


Fig 2

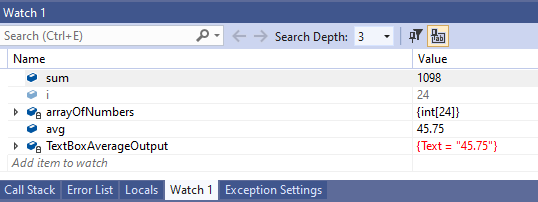


Fig 3

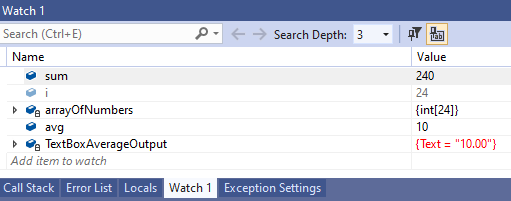


Fig 4

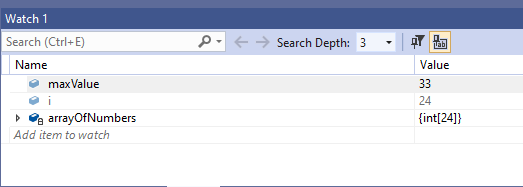


Fig 5

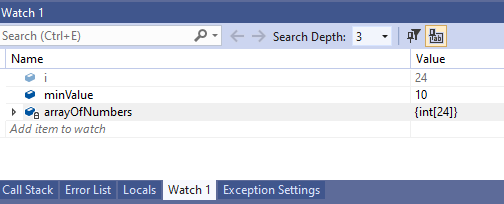


Fig 6

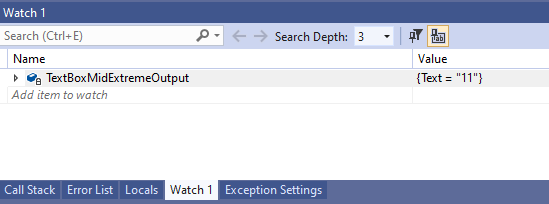


Fig 7

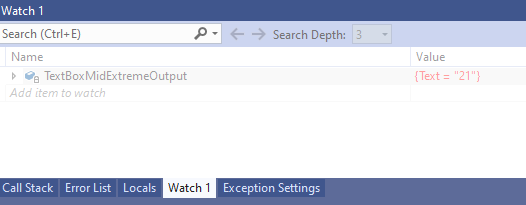


Fig 8

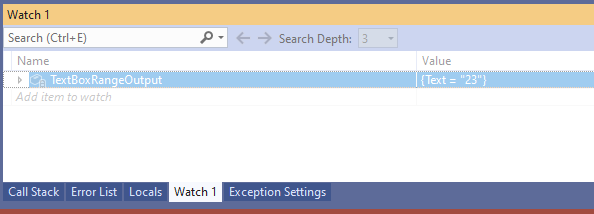


Fig 9

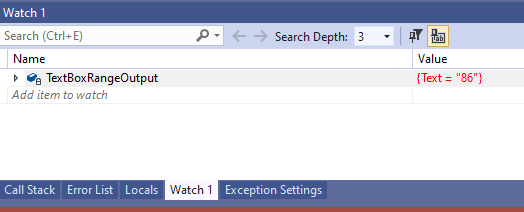


Fig 10

Graphical user interface

Description automatically generated

Fig 11

Background pattern

Description automatically generated

Fig 12

Graphical user interface, background pattern

Description automatically generated with medium confidence

Fig 13