|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Qualification details** | | | | | | | | | | | | | | | | |
| **Training Package Code and Title:** | | ICT - Information and Communications Technology (Release 8.1) | | | | | | | | | | | | | | |
| **Qualification National Code and Title:** | | ICT40120 Certificate IV in Information Technology  (Release 4) | | | | | | | | | | **State code:** | | | | BFF9 |
| **Assessment Title** | | Assessment Task Three Team Project | | | | | | | | | | | | | | |
| **Unit National Code & Title** | | ICTPRG440 Apply introductory programming skills in different languages | | | | | | | | | | | | | | |
| ICTPRG437 Build a user interface | | | | | | | | | | | | | | |
| ICTICT435 Create technical documentation | | | | | | | | | | | | | | |
| **Due Dates** | | Sprint One: Week 14 | | | | | | **Date Received** | | | | | | |  | |
| Sprint Two: Week 17 | | | | | | **Date Received** | | | | | | |  | |
| Handover: Week 19 | | | | | | **Date Received** | | | | | | |  | |
| **Student Name** | |  | | | | | | | | | **Student ID** | | | | |  |
| **Student Declaration** | | I declare that the evidence submitted is my own work:  ………………………………………….. | | | | | | | | | | | | | | |
| **Assessor Name** | |  | | | | | | | | | | | | | | |
| **Assessment Decision** | |  | Satisfactory | | | |  | | Not Yet Satisfactory | | | | | | | |
| **Assessor Signature** | |  | | | | | | | **Date** | | | | |  | | |
| **Is student eligible for reassessment (Re-sit)?** | |  | | No |  | Yes | | | **Reassessment Date:** | | | | | Week Twenty | | |
| **Feedback to student** | | | | | | | | | | | | | | | | |
| *Via Blackboard (LMS) – Please check [Grade] section.* | | | | | | | | | | | | | | | | |
| **Feedback from student** | | | | | | | | | | | | | | | | |
| *Via Blackboard (LMS) – Please use [Comment] section during submission.* | | | | | | | | | | | | | | | | |
| **Student signature** |  | | | | | | | | | **Date** | | |  | | | |

|  |  |
| --- | --- |
| **Assessment Instructions** | |
| **TO THE ASSESSOR** | |
| Type of Assessment | Team Project |
| Duration of Assessment | 8 Class Sessions (Week 12 - 19) |
| Location of Assessment | Classroom |
| Conditions | Assessor to ensure that the noise levels, natural interactions and time variances are maintained as it would be in the Software Development industry.  Learners are required to complete the required tasks in class and submit the required documentation electronically via Blackboard |
| Elements and Criteria | As detailed in the assessment plan  You are required to make sure that all students meet the elements, performance criteria and oral communication items as outlined in the provided checklist and marking guide. |
| Instructions | This project uses an agile methodology consisting of Sprint One, Sprint Two with a final submission of a Handover. Assessors need to ensure that students are in teams of two. They are required to select a Scrum Master for each sprint to lead the team and submit the documents required for each sprint (in Sprint One Question - 1, 2, 3…7 – Sprint Two - Question 4, 5, 6…14). Assessor must observe the students fulfilling their team responsibilities either as a Scrum Master or a team member using the Observation Checklist. In Sprint Two the roles will be reversed, and the Scrum Master will become the team member and vice-versa.  Students will need to demonstrate their workflow and code versioning by providing access to their GitHub account  In order to verify the authenticity of the student’s assessment, you may ask the student to again produce an answer to an existing question. |
| **TO THE STUDENT** | |
| Purpose of Assessment | You are required to show you can:  ICTPRG440 Apply introductory programming skills in different languages   * Demonstrate your skills and knowledge by creating a GUI based application * Code using data structures and standard algorithms for searching and sorting data. * Debug, document and test completed application using IDE and associated features.   ICTPRG437 Build a user interface   * Demonstrate your knowledge by researching prototyping tools and application development languages. * Investigating organizational guideline, policies and procedures.   ICTICT435 Create technical documentation   * Demonstrate your knowledge of technical document styles and design. * Investigate organisational policies, procedures and standards that cover document design. * Document scripts for internal and external stakeholders. * Collaborate and discuss ideas and requirements with team members.   The student must demonstrate the ability to complete the tasks outlined in this assessment and is expected to use systematic analytical processes and effect time management to meet the goals/deadlines outlined in the DAP.  You are required to meet the elements, performance criteria and oral communication items as outlined in the provided checklist. |
| Allowable Materials | Blackboard (Topic by topic) will include the following: Weekly Readings, Class notes, and Weekly Activities. |
| Required Resources | Computer with:   * Web links and example code can be downloaded from the Blackboard portal * MS Visual Studio, * MSOffice * Internet Access to MSDN, GitHub and www.citems.com.au/ |
| Reasonable Adjustment | In some circumstances, adjustments to assessments may be made for you. If you require support for literacy and numeracy issues; support for hearing, sight or mobility issues; change to assessment times/venues; use of special or adaptive technology; considerations relating to age, gender and cultural beliefs; format of assessment materials; or presence of a scribe you need to inform your lecturer. |
| Assessment Submission | All questions and activities must be attempted. All written answers must be submitted in this assessment document in the appropriate space.  Use of research tools and peers in formulating answers are acceptable – but work submitted must be your own work.  Final project documentation is to be uploaded to the appropriate area in the Blackboard course created for this unit.  If you are marked as NYS (Not Yet Satisfactory) on your first attempt, you will be provided with another opportunity to re-attempt the assessment. |
| Project contents | This team project consists of the following tasks:  Sprint One   * Question 1 – Complete the Project Specification, * Question 2 – Create a GUI design, * Question 3 – Algorithm and pseudo code, * Question 4 – Scrum board and meetings, * Question 5 – Sign off and approval, * Question 6 – Develop the Sprint One application, * Question 7 – Presentation of the completed Sprint One application and associated documentation.   Sprint Two   * Question 8 – Compete the Updated Project Specification, * Question 9 – Updated GUI design, * Question 10 – Algorithm and pseudo code, * Question 11 – Scrum board and meetings, * Question 12 – Sign off and approval, * Question 13 – Develop the Sprint Two application, * Question 14 – Presentation of the completed Sprint Two application and associated documentation   Handover   * Question 15 – Complete the Test Report * Question 16 – Recommendations * Question 17 – Review and handover |

## Software Development (Handover)

This team project will use an agile methodology to complete the development of the Astronomical Processing application. For detailed information about the agile software development methodology read the PDF documents in the Software Development section on Blackboard.

It is critical that all team members are familiar with all aspects of the development. There are three major milestones, the first two will be the assessment point which focuses on the Scrum Master (Sprint One and Sprint Two). The other team member will also submit but is not required to present the Sprint application or answer questions. The third milestone is for the whole team and represents the final assessment point for the project (Handover). The Weekly Schedule (show below) outlines each of these milestones and the final group handover. As the development progresses the team will collect information, create mini-reports and develop the Astronomical Processing application.

The team will utilise all the skills from previous assessment tasks and the example code and documentation on Blackboard.

## Updated Weekly Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| MILESTONE | | TASK | DESCRIPTION |
| Week 12  AM session | Sprint One |  |  |
| PM Session | Sprint One |  |  |
| Week 13  AM session | Sprint One |  |  |
| PM Session | Sprint One |  |  |
| Week 14  AM session | Sprint One |  |  |
| PM Session | Sprint One |  |  |
| Week 15  AM session | Sprint Two |  |  |
| PM Session | Sprint Two |  |  |
| Week 16  AM session | Sprint Two |  |  |
| PM Session | Sprint Two |  |  |
| Week 17  AM session | Sprint Two |  |  |
| PM session | Sprint Two |  |  |
| Week 18  AM session | Handover | Question Fifteen.  Commence Testing | The team will conduct testing of the completed application. |
| PM Session | Handover | Question Fifteen.  Compete Testing Report | The team will complete the test report. |
| Week 19  AM session | Handover | Question Sixteen.  Complete Recommendation Report | The team will review the application and document a list of improvement and recommendations. |
| PM Session | Handover | Question Seventeen.  **Assessment Point for all team members** | The team present the application and reports. The lecturer/assessor can ask questions of both team members. |

## Rapid Application Team

In Sprint One and Sprint Two each team member assumed the role of Scrum Master and Team Member; in this final task all team members are equal and share the responsibility to complete all the assessment criteria.

You should consult with the CITEMS representative (Your Lecturer) if you are unsure about any of the problems or questions in this assessment. Your primary research should focus on the resources on the Blackboard website, additional information can be collected from the Internet, ensure all sources are fully referenced. You should write your answers in one of the standard templates provided on Blackboard.

# Handover

The client has accepted the sprint one and sprint two development milestones for the Astronomical Processing application. The final stage is to conduct testing on all methods that calculate the four mathematic functions. This is to validate the accuracy of the methods.

The client wants the tests recorded in a formal Test Report with associated screen captures of the debug session which traces the changing values for each of the four functions. Use the following client requirements and complete the Test Report by recording the testing of the four mathematical methods.

Finally, as part of the service the team will need to complete the improvement and recommendations report. This will indicate what is possible if the application was to be put into full production

### Client Requirements

* All four mathematical methods are tested (mean, mode, average and range).
* The results of the tests are recorded in the formal Test Report.
* Each mathematical function is tested more than three times.
* Each mathematical method has a break point.
* Each mathematical method has local variables displayed in a watch.
* Recommendations report.

### Testing

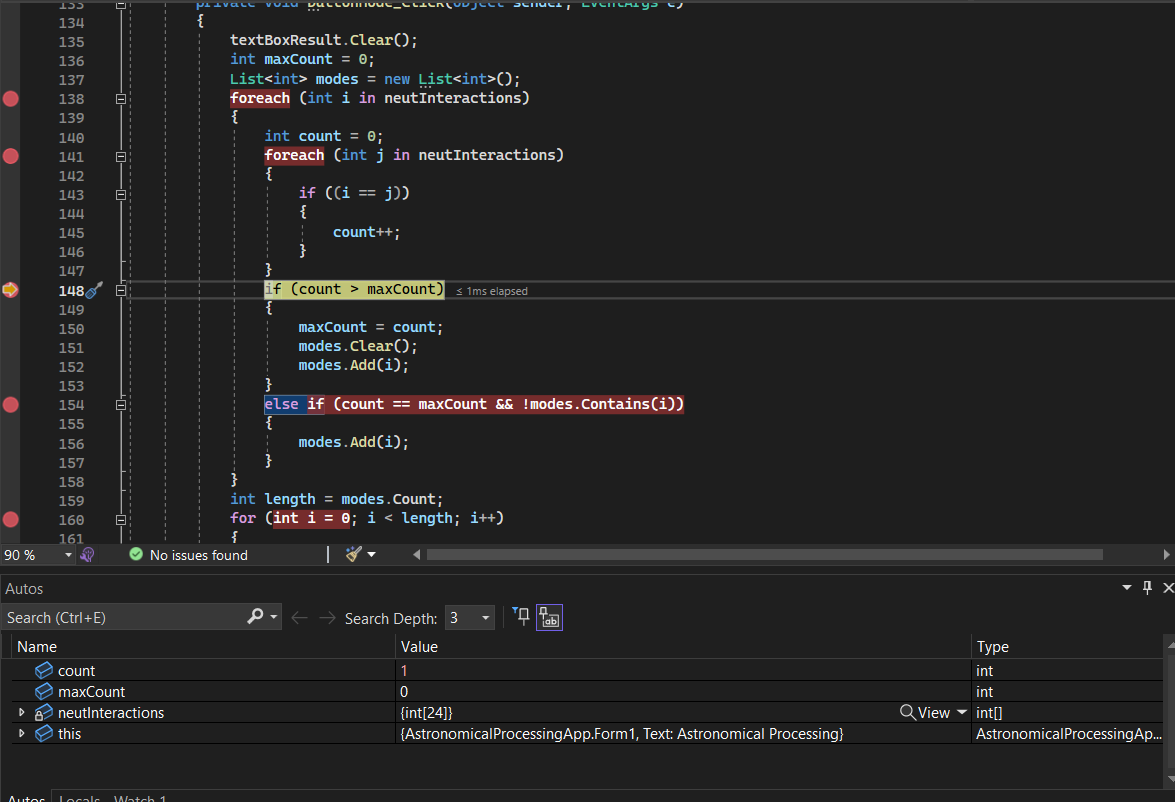
Read the Client Requirements and ensure all relevant information is included in the Test Report. This is a team effort so the team should meet and discuss how each of the four mathematical functions can be fully tested. The meeting should consider what type of data will be required for each mathematical function and reflect the organisational guidelines of CITEMS (refer www.citems.com.au/). Use the Test Report during the test session, add additional rows as required and include suitable screen captures to support each test case. Where the testing highlights an issue in the Astronomical Processing application you can updated the code and record this change to ensure all client requirements and testing are satisfactory.

## 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 | | | | | | | | |
| Date | | 02/06/2023 | | | | | | |
| Team Name | | JJ | | | | | | |
| Team Members | | James Chellew | | | Joshua Barker | | | |
| Test Type | | **Testing Mode** | | | | | | |
| Test Case # | Test Case Name | | Test Steps | Test Data | | Expected Results | Evidence  (ref to Screen Capture) | Pass / Fail |
| 1 | Normal Conditions | | Run app, press mode button. |  | | 11 |  | pass |
| 2 | Multiple Modes | | 1 – Run App  2 – Modify data to include 2 Modes  3 – Press mode button |  | | 22 and 92 |  | Pass |
| 3 | Extreme case 1, make every number different. | | 1 – Run App  2 – sort data  3 – Change duplicate data to different number  4 – Press mode button |  | | All numbers separated by commas |  | pass |
| 4 | Extreme case 2, large numbers and negatives | | 1 – Run app  2 – sort data  3 – add an assortment of negative and large numbers  4 – make a negative number the mode  5 – press mode button |  | | -1 |  | pass |

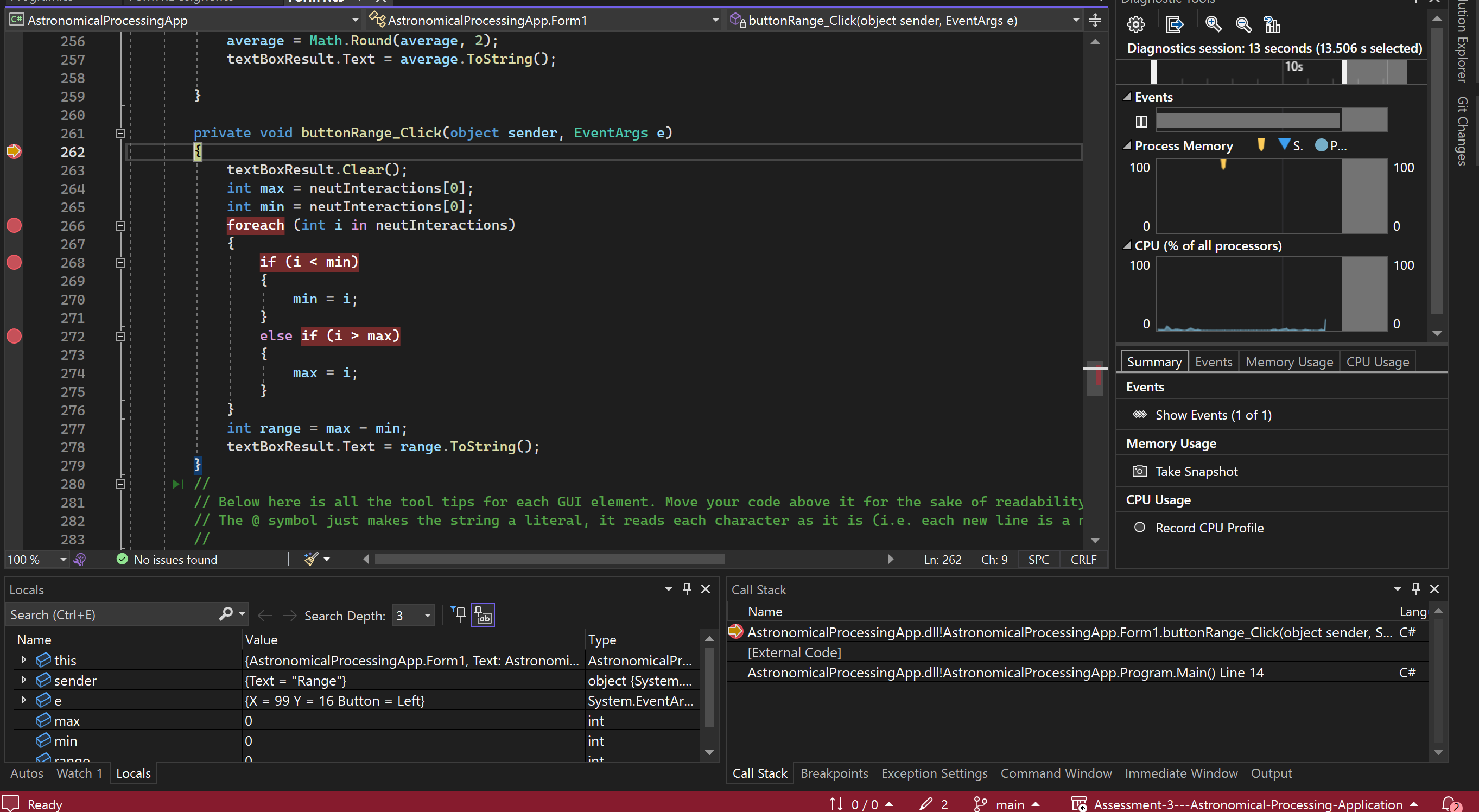


|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Report | | | | | | | | |
| Project Details | | | | | | | | |
| Date | | 02/06/2023 | | | | | | |
| Team Name | | JJ | | | | | | |
| Team Members | | James Chellew | | | Joshua Barker | | | |
| Test Type | | **Testing Mean (Average)** | | | | | | |
| Test Case # | Test Case Name | | Test Steps | Test Data | | Expected Results | Evidence  (ref to Screen Capture) | Pass / Fail |
| 1 | Normal Case, No sort | | Press mean. |  | | 48.29 |  | Pass |
| 2 | Normal Case, with sort | | Press sort.  Press mean. |  | | 44.38 |  | Pass |
| 3 | Large numbers + negative | | Press sort.  Add a few random large numbers (1000+).  Add a few negatives. |  | | 426.42 |  | Pass |
| 4 | All same | | Press sort.  Set all numbers to 1. |  | | 1 |  | Pass |

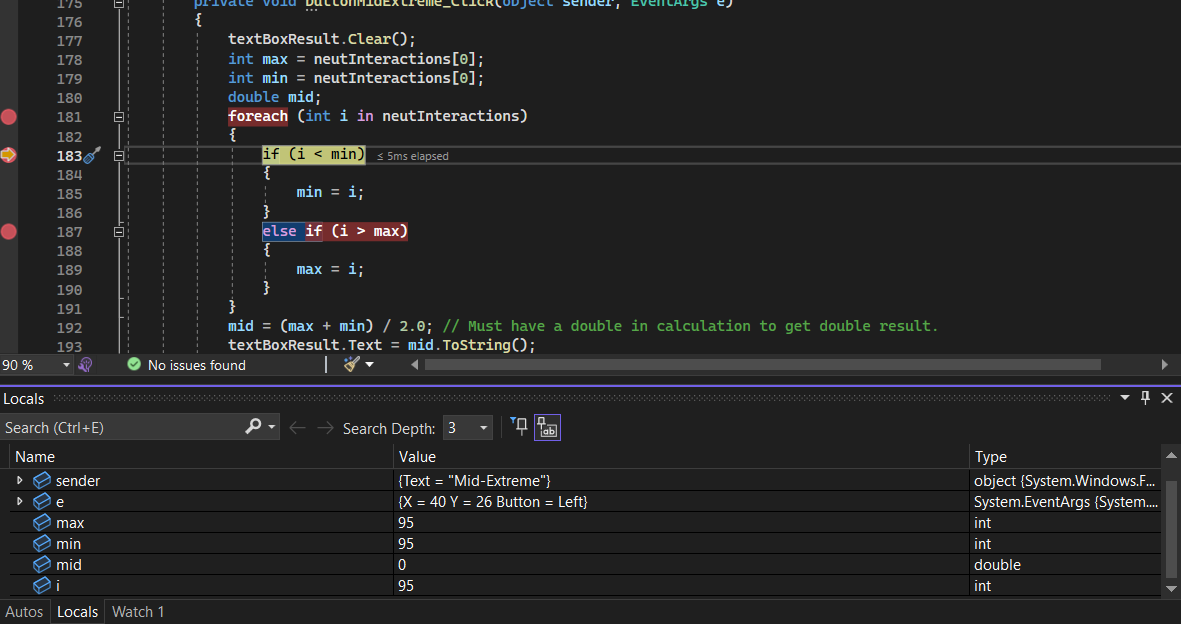
A screenshot of a computer program

Description automatically generated with medium confidence

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Report | | | | | | | | |
| Project Details | | | | | | | | |
| Date | | 02/06/2023 | | | | | | |
| Team Name | | JJ | | | | | | |
| Team Members | | James Chellew | | | Joshua Barker | | | |
| Test Type | | **Testing Range** | | | | | | |
| Test Case # | Test Case Name | | Test Steps | Test Data | | Expected Results | Evidence  (ref to Screen Capture) | Pass / Fail |
| 1 | Normal ConditionsNo sort | | 1 – Run app  2 – Press range |  | | 85 |  | Pass |
| 2 | Normal ConditionsWith Sort | | 1 – Run app  2 – Press sort  3 – Press range |  | | 82 |  | Pass |
| 3 | Large numbers | | 1 – Run app  2 – Press sort  3 – Modify min and max to –10000 and 10000 |  | | 20000 | 20000 | Pass |
| 4 | All same numbers | | 1 – Run app  2 – Press sort  3 – set all numbers to 1 |  | | 0 |  | Pass |



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Report | | | | | | | | |
| Project Details | | | | | | | | |
| Date | | 02/06/2023 | | | | | | |
| Team Name | | JJ | | | | | | |
| Team Members | | James Chellew | | | Joshua Barker | | | |
| Test Type | | **Testing Mid-Extreme** | | | | | | |
| Test Case # | Test Case Name | | Test Steps | Test Data | | Expected Results | Evidence  (ref to Screen Capture) | Pass / Fail |
| 1 | Normal Case | | 1 – run app  2 – press mid-extreme button  3 – press sort (for readability) |  | | 59 |  | pass |
| 2 | Sort before calculat-ing | | 1 – run app  2 – press sort (for readability)  3 – press mid-extreme button |  | | 53.5 |  | fail |
| 3 | Massive difference | | 1 – run app  2 – sort  3 – Change extremes to –10000 and 10000  4 – press mid-extreme |  | | 0 |  | pass |
| 4 | All numbers the same | | 1 – run app  2 – edit all numbers to 1  3 –press mid extreme |  | | 1 | (Changed method to produce double after test 2) | fail |
| 5 | Retest all same number | | 1 – run app  2 – edit all numbers to 1  3 –press mid extreme |  | | 1 |  | pass |



## Question Sixteen

Review the Astronomical Application and consider how this might be improved. This is a team effort so the team should meet and discuss how each of the components/features could be improved. The following items are provided as a starting point.

* What additional buttons/features could be included?
* What additional mathematical functions might be useful?
* What error trapping could be included?
* How could data be saved and retrieved?
* What menu bars could be used?

|  |  |  |  |
| --- | --- | --- | --- |
| Recommendation Report | | | |
| Project Details | | | |
| Date | | 02/06/2023 | |
| Team Name | | JJ | |
| Team Members | | James Chellew | Joshua Barker |
| # | Recommendation Details | | |
| 1 | Extra button to load new data | | |
| 2 | Change design to accommodate a wider range of numbers (interactions per minute as an example). Currently capped at 24 numbers in array | | |
| 3 | Would change the search function to select searched value if found. | | |
| 4 | Add a save data function – to txt file and a way to load said txt file (saved as the days date) | | |
| 5 | Add a second listbox to compare different sets of data. | | |
| 6 | Add median button. | | |
| 7 | Add mean button exclusive of extreme values. | | |

## Question Seventeen

The team should review all documentation and then notify your Lecturer to arrange a suitable time to review the Handover documents and the working Astronomical Processing application. This review will be assessing the both tam members. The following Marking Guide should be used to ensure all aspects of the assessment are covered.

## Submission Requirements

Your submission for the Handover will include:

Completed Test Report from Question Fifteen,

Completed Recommendations Report from Question Sixteen,

Completed Solution Folder for the Astronomical Processing application.

Your team will submit the working program and Test Report documentation to the appropriate Blackboard section. The following Marking Guide should be used to ensure all aspects of the final Handover assessment are covered. Consult your lecturer for further information or clarification.

|  |  |  |  |
| --- | --- | --- | --- |
| Assessment Task Three Handover  Marking Guide | | | |
| Student Name | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| Criteria | Satisfactory  Yes No | | Comment |
| 1. The test data is suitable for each test case and reflects the design and development requirements. |  |  |  |
| 1. The test results are support with screen captures. |  |  |  |
| 1. The Test Report is complete. |  |  |  |
| 1. The Recommendations Report is complete |  |  |  |
| 1. All documentation is properly formatted |  |  |  |
| 1. Participated in both sprints |  |  |  |
| 1. Participated in the testing/recommendations/handover |  |  |  |
| 1. Active and supportive team members |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Assessor Name** |  | | | | |
| **Assessment Decision** |  | Satisfactory |  | Not Yet Satisfactory | |
| **Is student eligible for reassessment (Re-sit)?** | No | Yes | **Reassessment Date:** | |  |
| **Assessor Signature** |  | | **Date** | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Feedback to student** | | | |
|  | | | |
| **Student signature** |  | **Date** |  |

**NOTE to Assessor: This is an individual submission. Please complete one checklist per student.**

End of Assessment