**Instructions on Assessment:**

Based on a .txt file of data provided to each student individually, create a c programme in Notepad++ which can do the following:

* Load this information from file
* Perform mathematical analysis on it (mean, median, range, st dev, best fit equation, differential/integral)
* Save the analysis to file
* Display a graph of said data (including axes, axes labels and a straight best fit line)
* The code should use a menu structure so that it is easy to navigate and the user should be able to change features and return to the menu without the programme ending.

The code will also be tested with a different .txt file of data, which will not be available to students (therefore the code should be adaptable to different scales of data).

|  |  |
| --- | --- |
| Achievement | Maximum Mark |
| **EXECUTION OF PROGRAM** | **(70)** |
| Program builds useful executable from source code   * Deductions for warnings * Zero if code does not compile * Is there a way of ‘breaking’ the code when it is running? | 10 |
| Running program meets requirements   * Loads information from file * Perform mathematical analysis on it (mean, median, range, st dev, best fit equation, differential/integral of best fit equation) * Save the analysis to file * Display a graph of said data – axes, labels, scaling * Best fit line | 50 |
| Good design of user interface   * Menu system   Use of menu, displaying current values correctly | 10 |
| **CODE STYLE** | **(30)** |
| Good layout, indentation | 5 |
| Good use of comments to clarify intent   * Good use of comments which explain fully what is going on * Not overly detailed * Comments must complement code | 5 |
| Code is easy to follow and maintain, simple design   * Use of functions and header files to improve code clarity * No obscure use of code * Sensible variable and function names * Appropriate use and definition of parameters and functions where applicable * Well-structured code | 20 |

The marks above for each section will be determined by how well the requirement for each is met.

**Additional Instructions to students:**

Marks will be awarded for good program structure and a clear report showing understanding of the main concepts (See assessment criteria.)

The program should be tested with suitable test data and compared with theoretical calculations to prove that it is correct.

**Further Information**

**Learning Outcomes assessed in this assessment:**

• Develop a program from a specification and provide suitable documentation.

• Develop a C program for a given task and prove that the program will work correctly through the use of the Debug tool.

• Make effective use of the C built in software libraries.

**Expected size of the submission:**

This should be apparent from the size of the program and also see specifications.

**Referencing Style:**

Not required

**ASSESSMENT REGULATIONS**

You are advised to read the guidance for students regarding assessment policies. They are available online [here](https://www.northumbria.ac.uk/about-us/university-services/academic-registry/quality-and-teaching-excellence/assessment/guidance-for-students/).

**Late submission of work**

Where coursework is submitted without approval, after the published hand-in deadline, the following penalties will apply.

For coursework submitted up to 1 working day (24 hours) after the published hand-in deadline without approval, **10% of the total marks available for the assessment** (i.e.100%) **shall be deducted** from the assessment mark.

Coursework submitted more than 1 working day (24 hours) after the published hand-in deadline without approval will be regarded as not having been completed. **A mark of zero will be awarded for the assessment and the module will be failed**, irrespective of the overall module mark.

These provisions apply to all assessments, including those assessed on a Pass/Fail basis.

The full policy can be found [here](https://northumbria-cdn.azureedge.net/-/media/corporate-website/new-sitecore-gallery/services/academic-registry/documents/qte/assessment/guidance-for-students/late-submission-of-work-and-extension-requests-policy.pdf?la=en&modified=20181005134740&hash=AF8200F74AF421F9F19E5332B8FD3F3D8E116292).

**Word limits and penalties**

If the assignment is within +10% of the stated word limit no penalty will apply.

The word count is to be declared on the front page of your assignment and the assignment cover sheet. The word count does not include:

eg appendices, glossary, footnotes, tables

Please note, in text citations [e.g. (Smith, 2011)] and direct secondary quotations [e.g. “*dib-dab nonsense analysis*” (Smith, 2011 p.123)] are INCLUDED in the word count.

***Students must retain an electronic copy of this assignment (including ALL appendices) and it must be made available within 24hours of them requesting it be submitted.***

**Academic Misconduct**

The Assessment Regulations for Taught Awards (ARTA) contain the ***Regulations and procedures applying to cheating, plagiarism and other forms of academic misconduct***.

You are reminded that plagiarism, collusion and other forms of academic misconduct as referred to in the Academic Misconduct procedure of the assessment regulations are taken very seriously. Assignments in which evidence of plagiarism or other forms of academic misconduct is found may receive a mark of zero.