|  |  |
| --- | --- |
| **Validation of data** | |
| **Test** | **Result (pass / not pass)** |
| Format code of Student: STxxx | **Passed** |
| Format code of Professor: PRxxx | **Passed** |
| Name of Person has single space between words | **Passed** |
| Date of student must be valid | **Passed** |
| Grade of student must be in range [0, 9] | **Passed** |
| Student codes are unique | **Passed** |

|  |  |
| --- | --- |
| **Function** | |
| **Method** | **Result** |
| Load initial data from file ‘initial data.txt’ | **Success** |
| Add new Student  (Option 1 → 1)   * code: ST020 * name: Henry Watson * address: 221B Baker Street, London, England * grade: 8 * day: 1 * month: 1 * year: 2015   *Note: Choose 5 (display) and then choose 1 to display result* | **Success** |
| Add students from file  (Option 1 → 2)   * file name: in\_students.txt   *Note: Choose 5 (display) and then choose 1 to display result* | **Success** |
| Remove a student  (Option 2 → 1)   * student code: ST008   *Note: Choose 5 (display) and then choose 1 to display result* | **Success** |
| Remove all invalid students  (Option 2 → 2)  *Note: Choose 5 (display) and then choose 1 to display result* | **Success** |

|  |  |
| --- | --- |
| Update student  (Option 3)   * code: ST020 * name: Henry J. Watson * address: skip * grade: 9 * day: 10 * month: 10 * year: skip   *Note: Choose 5 (display) and then choose 1 to see result* | **Success** |
| Find student  (Option 4)   * code: ST020   → Result: information of student ST020 | **Success** |
| Display students sorted code ascending  (Option 5 → 1) | **Success** |
| Display students sorted name ascending  (Option 5 → 2) | **Success** |
| Display students sorted grade descending  (Option 5 → 3) | **Success** |
| Display highest grade students  (Option 5 → 4)  → Result: Two students with grade 9.9 | **Success** |
| Display lowest grade students  (Option 5 → 5)  → Result: Students ST009 with grade 2.0 | **Success** |
| Display average grade  (Option 6 → 1)  → Result: 6.42, suppose that all the above tests were executed. | **Success** |
| Display distribution of grade  (Option 6 → 2) | **Success** |
| Display distinct grade  (Option 6 → 3) | **Success** |
| Display duplicate student names  (Option 6 → 4) |  |
| Display unique student names  (Option 6 → 5) | **Success** |
| Export student list to file  (Option 7)   * file name: out\_students.txt | **Success** |
| Update professor  (Option 8 → 1)  *Note: Fist, choose 8 (about professor) and then choose 2 (display professor) to see the difference*   * name: skip * address: skip * position: HONOR\_PROFESSOR * edu: DOCTOR * experience: 30 * basic salary: 3000 | **Success** |
| Display professor  (Option 8 → 2) | **Success** |
| Display students with grade higher than 5  (Option 9 → 1) | **Success** |
| Display students with grade lower than 5  (Option 9 → 2) | **Success** |
| Display n-th year student  (Option 9 → 3)   * n: 5   → Result: display students: ST001, ST002, ST006, ST007 *(suppose that all above tests were excuted)* | **Success** |
| Display number of students enrolled in a year  (Option 9 → 4)   * year: 2016   → Result: 3 students | **Success** |

Other details: this program is able to handle invalid input data from user to avoid making program crash.