Candidate name: ..................................................

Centre number: ..................................................

Index number: ..................................................

Programming language used: .....................................................................

|  |
| --- |
| **Question 1** |
| **Evidence 1**  *Program code*  def main():  print('1. Read file data')  print('2. Bubble sort')  print('3. Quick sort / Insertion sort')  print('4. End')  option = int(input('Enter your option: '))  while option != 1 or option != 2 or option != 3 or option != 4:  print('Invalid option. Please pick an option from 1 to 4.')  option = int(input('Enter your option: '))  admissions = []  if option == 1:  with open('ADMISSIONS-DATA.TXT') as file:  for line in file:  num = int(line.strip())  admissions.append(num)  main() |
| **Evidence 2**  *Program code*  swapped = True          end = len(admissions)          while swapped:              swapped = False              for i in range(1, end):                  if admissions[i - 1] > admissions[i]:                      tmp = admissions[i]                      admissions[i] = admissions[i - 1]                      admissions[i - 1] = tmp                      swapped = True              end -= 1 |
| **Evidence 3**  The sort algorithm used was :……insertion sort………  *Program code*  # Sort using insertion sort          for i in range(1, len(admissions)):              cur = admissions[i]              j = i              while j > 0 and admissions[j - 1] > cur:                  admissions[j] = admissions[j - 1]                  j -= 1              admissions[j] = cur |
| **Evidence 4**  *Screenshot* |
| **Evidence 5**  *Screenshot* |
| **Question 2** |
| **Evidence 6**  *Program code*  *Screenshot* |
| **Evidence 7** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Evidence 8**  *Amended Program Code*   |  |  |  | | --- | --- | --- | | DenaryNumber | Purpose of the test | Expected  Output | |  |  |  | |  |  |  | |  |  |  |   *Screenshot 1*  *Screenshot 2* |
| **Question 3** |
| **Evidence 9**  *Program code*  *Screenshots X 3* |
| **Evidence 10**  *Program code*  *Screenshots X 2* |
| **Evidence 11**  *Program code*  *Screenshots X 3* |
| **Evidence 12**  *Program code* |

|  |
| --- |
| **Question 4** |
| **Evidence 13**   |  |  |  |  | | --- | --- | --- | --- | | Test Number | User ID | Return value | Explanation of the test case | | 1 | 2015\_0987 | 0 | Valid User ID | | 2 |  |  |  | | 3 |  |  |  | | 4 |  |  |  | |
| **Evidence 14**  *Program code*  *Screenshots X 3* |
| **Evidence 15**  *Program code* |
| **Evidence 16**  *Program code* |
| **Evidence 17**  *Program code X 2* |
| **Evidence 18**  *Program code*  *Screenshot* |
| **Evidence 19**  *Program code*  *Screenshot* |