**OGUNGBAYI IMRAN**

**EEG/2022/103**

**OBJ**

1. **B** – C is compiled while Python is interpreted. Compiled languages are generally faster than interpreted languages because they convert the code in bulk unlike interpreted ones.

2. **C** – Moores law states that the not of transistors on a chip doubles roughly every 2 years leading to an exponential growth in computing power.

3. **D** – A compiles the code into Assembly(Compiler) while B converts it to Machine language(Assembler)

4. **A** – The code is written in C, compiled to Assembly and assembled to Machine Language.

5. B **–** The response time remains the same

6. **D** – Imperative thinking is known as ‘how-to’ knowledge.

7. **B** – Turing’s law states that any function that can be algorithimized can be compued by any device or language

8. **C** – Python has modules and libaries that support a waide range of field spanning from ML, Web Dev, Gaming, e.t.c

9. **C** – A while loop runs infinitely as long as the test case is met (true)

10. **B** – Code snippet A still tests if even numbers are divisors, which is redundant, compared to snippet B that excludes even numbers

11. **B** – Lists are mutable ( can be edited after being declared) while tuples are not

12. D – O[logn] – bisection complexity is logn

13. **D** – A line graph is best for visualizing continuos data such as one from a sensor

14. **C** – The decision making blocks are not structured properly, they’re meant to be conditional paths not straight forward

15. **A** – For the square root to be nearest o the answer the epsilon mst be very small maybe 0.000001

16. **D** - ‘==’ is an equality comparison operator

17. **B** – ‘clear’ is used to clear the workspace ( data and variables), ‘clc’ is just for clearing the command line.

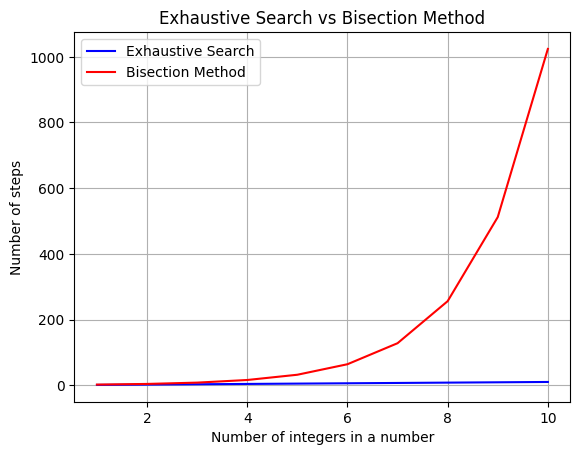
18. **D** – Simulink is more suited because it allows easier implementation of this task

19. D – correct definition of matrices and proper syntax for division

20. **B** – recursion is best for cleanly looping through the process of tetrating

**THEORY**

1. The relationship between Binary and exhaustive search is more like a n and 2n relationship



2. #CommentA – If the power is even and the number is a negative integer, return None(No answer)

#CommentB – Set the initial guess to the middle of the initial search pool(binary search)

#CommentC – Loop as long as the difference between the square of the guess and the answer is larger than epsilon

#CommentD – Set the lower boundary of the search pool as the smaller value between the number and -1

#CommentE – Set the new guess to the middle of the new search pool(binary search)

3.

a. x = 3

b. y = [ 8

2

4]

c. z = [ 1 2 4 ]

d. q = [ 5 2 6 ]

e. k = [ 12 17 3

5 8 3

1 2 4

2 4 6 ]

4. Pseudo code

1. Take in the word
2. Check recursively the nth and (length of the word – n)th word , if any check is false declare the word as not a palindrome else continue e.g (first and last letter, second and second to last letter)
3. If all check is true , declare the word as palindrome