|  |  |  |  |
| --- | --- | --- | --- |
| **NAME:** | Lauron, John Enrico D. | **DATE:** | 10/09/2023 |

ALGORITHM EXERCISE # 5.1

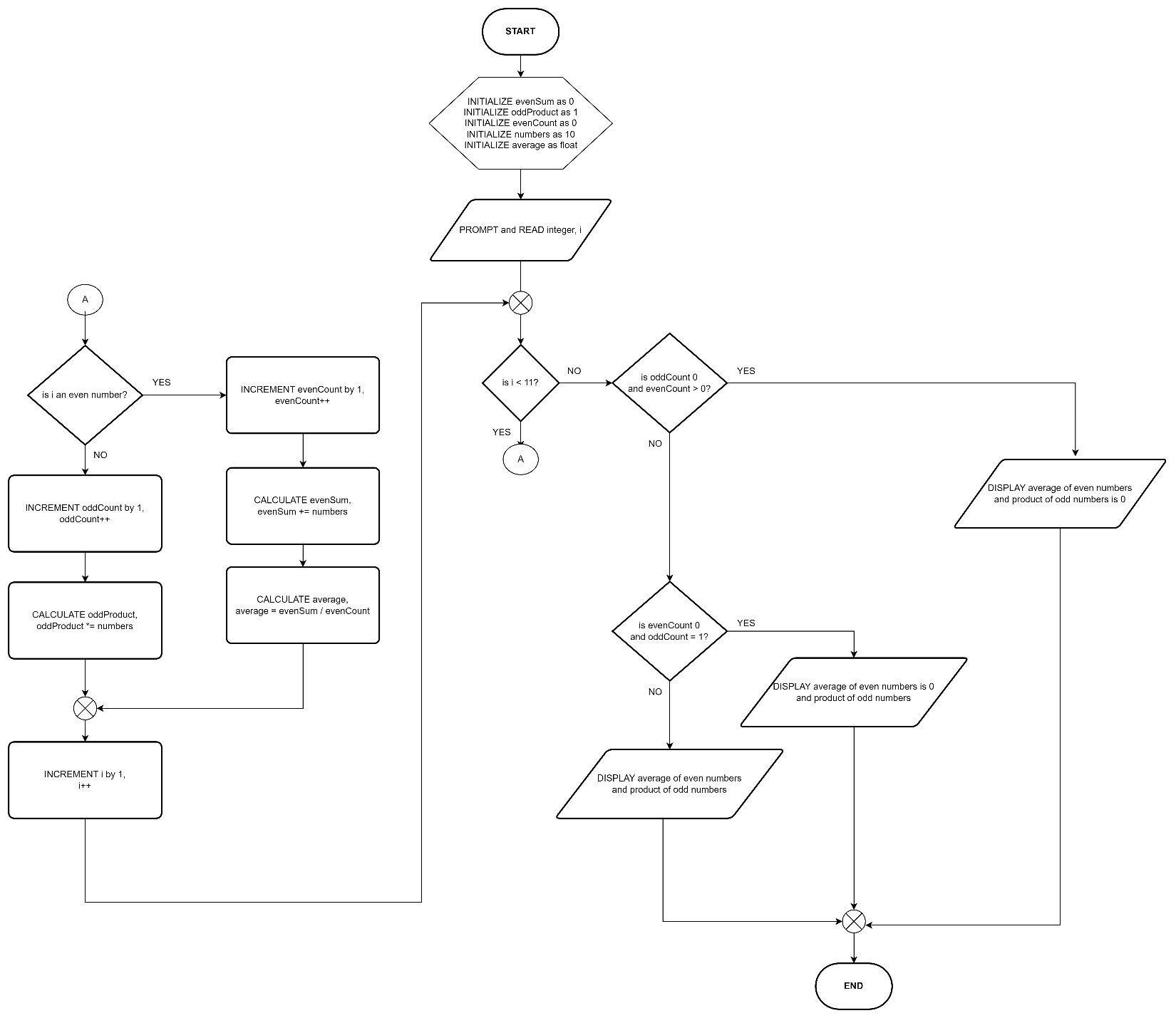
LE 5.11: **Average of Even Numbers and Product of Odd Numbers (Pseudocode)**

START

1. INITIALIZE evenSum as 0
2. INITIALIZE oddProduct as 1
3. INITIALIZE evenCount as 0
4. INITIALIZE numbers as 10
5. INITIALIZE average as float
6. PROMPT and READ integer, i
7. FOR i = 1; i < 11; i++ DO
   1. DISPLAY input integers, numbers [ i ]
8. FOR int i = 1; i < 11; i ++ DO
   1. IF numbers [ i ] is divisible by 2 THEN
      1. FIND the sum of the even numbers, evenSum
      2. evenCount++
   2. ELSE
      1. FIND the product of the odd numbers, oddProduct \*= numbers [ i ]
   3. ENDIF
9. ENDFOR
10. COMPUTE the average of all even numbers, evenSum / evenCount
11. IF oddProduct is equal to 1 and evenSum is greater than 0
    1. DISPLAY the average of all even integers
    2. DISPLAY the product of all odd integers is 0
12. ELSE IF oddProduct is not equal to 1 and evenSum is equal to 0
    1. DISPLAY the average of all even integers is 0
    2. DISPLAY the product of all odd integers
13. ELSE
    1. DISPLAY the average of all even integers and the product of all odd integers
14. ENDIF

END

LE 5.11: **Average of Even Numbers and Product of Odd Numbers (Flowchart)**

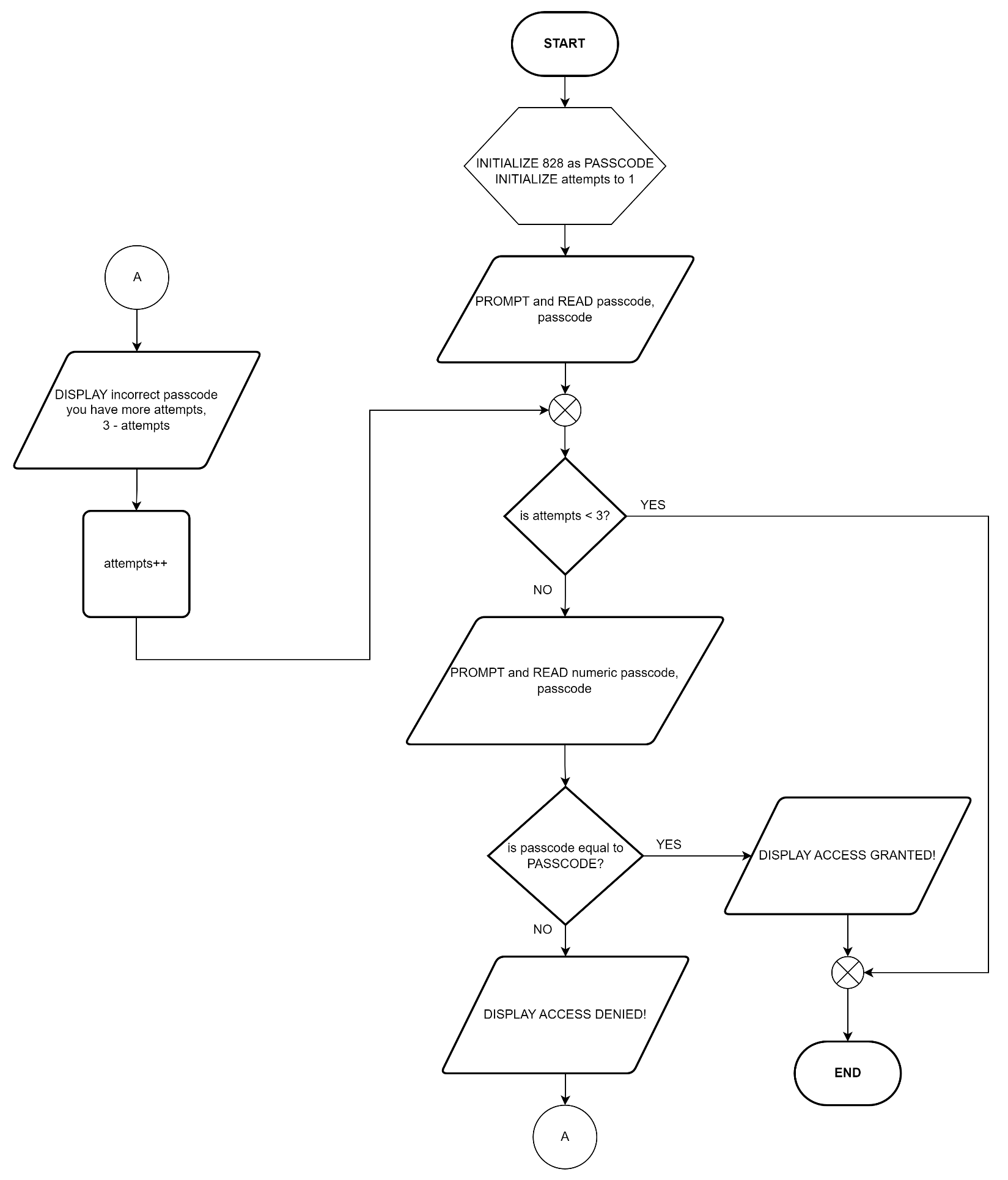
****

LE 5.12: **Passcode Lock (Pseudocode)**

START

1. INITIALIZE 828 as PASSCODE
2. INITIALIZE attempts to 1
3. PROMPT and READ passcode
4. WHILE passcode is not equal to PASSCODE and attempts is less than 3
   1. DISPLAY incorrect passcode you have more attempts, 3 – attempts
   2. attempts++
   3. PROMPT and READ the numeric passcode
5. IF passcode is equal to PASSCODE THEN
   1. DISPLAY ACCESS GRANTED
6. ELSE
   1. DISPLAY ACCESS DENIED
7. ENDIF
8. ENDWHILE

END

****LE 5.12: **Passcode Lock (Flowchart)**

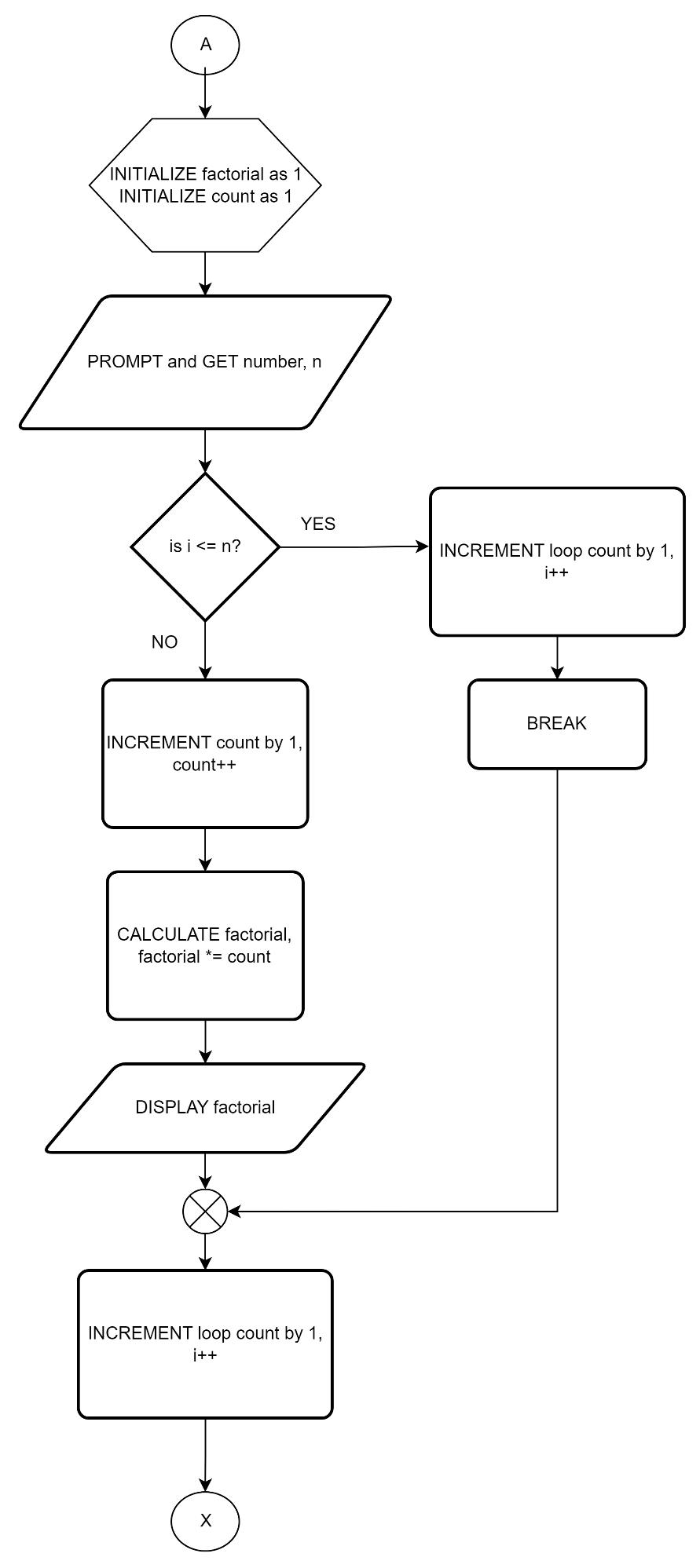
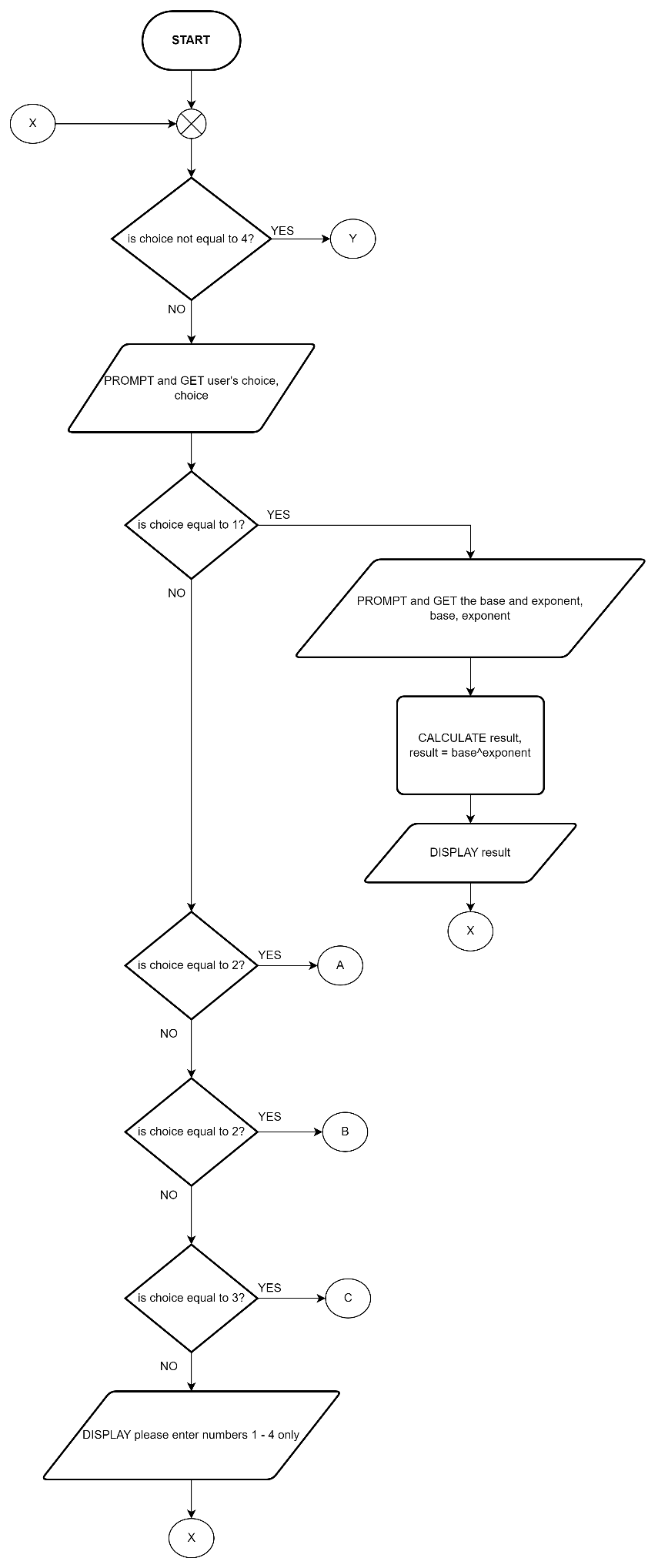
LE 5.13: **Problem Solver Menu (Pseudocode)**

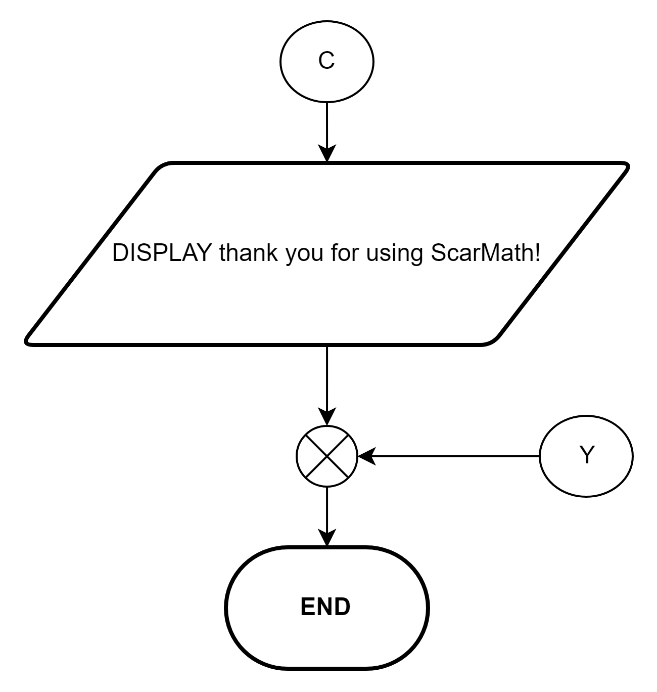
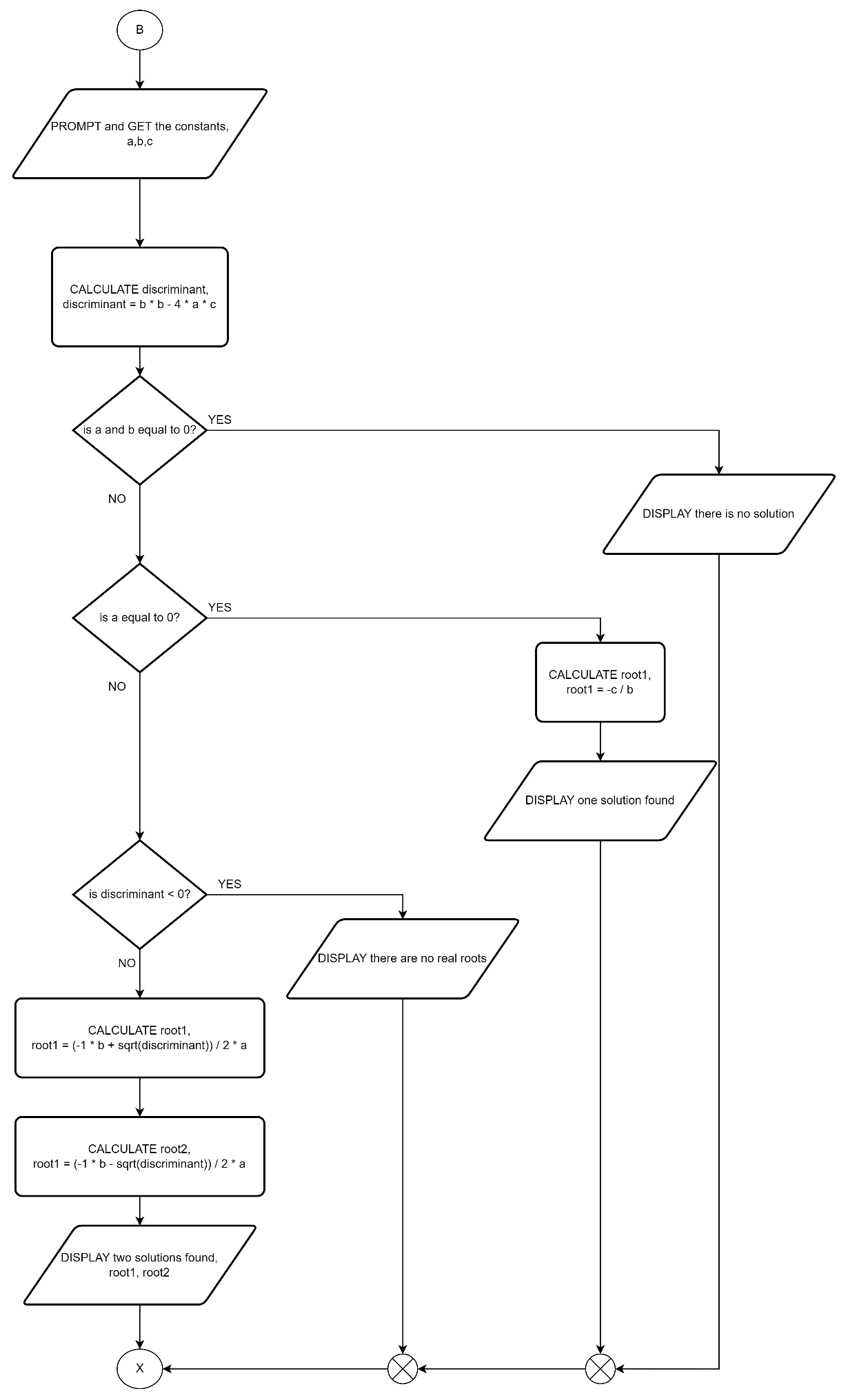
START

1. DO
   1. PROMPT and GET the user’s choice for the problem solvers, choice
   2. IF choice is equal to 1 THEN
      1. PROMPT and READ base, base
      2. PROMPT and READ exponent, exponent
      3. CALCULATE result, result = base^exponent
      4. DISPLAY result
   3. ELSE IF choice is equal to 2 THEN
      1. INITIALIZE factorial as 1
      2. PROMPT and READ number, n
      3. FOR count = 1; count <= n; i++ DO
      4. CALCULATE factorial, factorial \*= count
      5. DISPLAY factorial
2. ENDFOR
3. ELSE IF choice is equal to 3 THEN
   1. INITIALIZE a,b,c,discriminant,root1,root2 as float
   2. PROMPT and READ the value of a, c
   3. PROMPT and READ the value of b, c
   4. PROMPT and READ the value of c, c
   5. CALULATE discriminant, discriminant = b \* b – 4 \* a \* c
4. IF (a and b is equal to 0) THEN
   1. DISPLAY there is no solution
5. ELSE IF (a is equal to 0) THEN
   1. CALCULATE root1, root1 = -1 \* c / b
   2. DISPLAY there is only one root
6. ELSE IF (discriminant is less than 0) THEN
   1. DISPLAY there are no real roots
7. ELSE
   1. CALCULATE root1, root1 = (-1 \* b + sqrt(discriminant)) / (2 \* a)
   2. CALCULATE root1, root1 = (-1 \* b - sqrt(discriminant)) / (2 \* a)
   3. IF (root1 is equal to root2)
      1. DISPLAY one solution found
   4. ELSE
      1. DISPLAY two solutions found
8. ENDIF
   1. ELSE IF choice is equal to 4 THEN
      1. DISPLAY thank you for using ScarMath
   2. ELSE
      1. DISPLAY please enter numbers 1 – 4 only
9. WHILE choice is not equal to 4

END

LE 5.13: **Problem Solver Menu (Flowchart)**

****

****