|  |  |  |  |
| --- | --- | --- | --- |
| **NAME:** | Lauron, John Enrico D. | **DATE:** | 11/21/2023 |

ALGORITHM EXERCISE # 7

LE 7.11: **Largest and Smallest**

**main**

START

1. INITIALIZE numbers to 10, numbers[10]
2. INITIALIZE limit to 10
3. CALL module, readNumbers(numbers, limit)
4. CALL module, largestNumber = findLargestNumber(numbers, limit)
5. CALL module, smallestNumber = findsmallestNumber(numbers, limit)
6. CALL module, displayInputtedNumbers(numbers, limit)
7. DISPLAY the smallest number
8. DISPLAY the largest number

END

**readNumbers(int numbers[10], int limit)**

START

1. DISPLAY ENTER 10 NUMBERS!
2. FOR i = 0 to limit by 1 THEN
   1. READ a number, numbers[i]
3. ENDFOR

RETURN main

**findLargestNumber(int numbers[10], int limit)**

START

1. INITIALIZE largest = numbers[0]
2. FOR i = 0 to limit by 1 THEN
   1. IF(numbers[i] > largest) THEN
      1. SET largest = numbers[i]
   2. ENDIF
3. ENDFOR

RETURN largest

**findSmallestNumber(int numbers[10], int limit)**

START

1. INITIALIZE smallest = numbers[0]
2. FOR i = 0 to limit by 1 THEN
   1. IF(numbers[i] < smallest) THEN
      1. SET smallest = numbers[i]
   2. ENDIF
3. ENDFOR

RETURN smallest

**displayInputtedNumbers(int numbers[10], int limit)**

START

1. DISPLAY the numbers you entered are:
2. FOR i = 0 to limit by 1 THEN
   1. DISPLAY numbers[i]
3. ENDFOR
4. DISPLAY newline

RETURN main

LE 7.12: **Vowels and Consonants in String**

**main**

START

1. INITIALIZE char str[256]
2. INITIALIZE vowelCount, consonantCount
3. PROMPT and READ a string
4. CALL module, vowelCount = countVowels(str)
5. CALL module, consonantCount = countConsonants(str)
6. DISPLAY There are \_ vowels and \_ consonants in the string, vowelCount, consonantCount

END

**isVowel(char ch)**

START

RETURN (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ||

ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U')

**countVowels (char str[256])**

START

1. INITIALIZE vowelCount = 0
2. FOR i = 0 to str[i] by 1 THEN
   1. IF(CALL module, isVowel(str[i])) THEN
      1. INCREMENT vowelCount by 1
   2. ENDIF
3. ENDFOR

RETURN vowelCount

**countConsonants (char str[256])**

START

1. INITIALIZE consonantCount = 0
2. FOR i = 0 to str[i] by 1 THEN
   1. IF(str[i] >= 65 && str[i] <= 90 || str[i] >= 97 && str[i] <= 122) THEN
      1. IF(!CALL module, isVowel(str[i])) THEN
         1. INCREMENT consonantCount by 1
      2. ENDIF
   2. ENDIF
3. ENDFOR

RETURN consonantCount

LE 7.13: **Longest String**

**main**

START

1. INITIALIZE char str[5][256]
2. INITIALIZE numStrings = 5
3. CALL module, readStrings(str, numStrings)
4. CALL module, longestStringIndex = findLongestString(str, numStrings)
5. DISPLAY The longest string is:
6. PUTS (str[longestStringIndex])

END

**readStrings(char str[][256], int numStrings)**

START

1. FOR i = 0 to numStrings by 1 THEN
   1. READ a string, str[i]
2. ENDFOR

RETURN main

**findLongestString(char str[][256], int numStrings)**

START

1. INITIALIZE longestStringLength = 0
2. INITALIZE longestStringIndex = 0
3. FOR i = 0 to numStrings by 1 THEN
   1. INITIALIZE length = strlen(str[i])
   2. IF(length > longestStringLength) THEN
      1. SET longestStringLength = length
      2. SET longestStringIndex = i
   3. ENDIF
4. ENDFOR

RETURN longestStringIndex