

Comp 281 Assignment 1 Report

1014: Area/Circumference Sum of Circles

The program takes two integer inputs which are the radii of two circles. The program then loops for the amount of times that is the difference between the two inputs, using the first input ('cirOne') as the base for the loop (absolute value). The current value for 'areaSum' is its current value plus ($\pi * \text{cirOne} * \text{cirOne}$). The current value for 'circumSum' is its current value plus ($\text{cirOne} * (2 * \pi)$). 'cirOne' is incremented afterwards. Prints the areaSum and circumSum once the loop finishes.

1017: Product of Integers in Columns

Program takes an initial integer from the user, indicating the amount of columns to follow. Two arrays are created – one to store the number of entries per column and the product of these entries per column. A loop is made to obtain the quantity of values to go into the columns. A second for loop is used to occupy the product-holding array ('entries' in the program); a modulus operation is used to work out the current column, and this is used in an if statement to check the value does not equal 0 – if it does then the loop skips a step as that column no longer takes anymore entries. When the 'if' statement outputs true, the current column is multiplied by the input. Finally, a third loop produces the output by going through the 'entries' array.

For the sample input provided, the program seems to ignore the final input – this may be due to the 'for' loop using the total amount of entries required (in the sample case, 12).

1025: HCF and LCM of Two Integers

Program takes two integer inputs. A for loop is used, which runs 'inputOne' amount of times and an if statement checks if the modulus of inputOne and the 'loopVar' as well as inputTwo and loopVar both equal zero. If so, loopVar's value is saved into the 'hcf' variable. When the for loop finishes, hcf is printed to console. For the LCM, a while loop is used which reiterates whenever 'multOne' and 'multTwo' are not equal to each other. If statements are used to check which variable is higher and then increases the value of the lower by its corresponding input number. Break statements are used within the successive if statements which checks if changing the value still makes the while condition true. Program finally outputs multOne when multOne and multTwo equal each other.

1030: Precise Division

Program takes 3 unsigned integer inputs – 'a,' 'b,' and 'n' – where the result is a/b and outputs the 'n'th decimal place. Once the division has occurred, using float type casting for the integer inputs, a for loop multiplies the number by 10 with 'n' repetitions. Finally this value is type casted as an int and modulo ten is applied. This provides required output, which is sent to the console.

1060: Sorting Even Integers

Input is ten integers, stored in an array. A second array is created which stores the even numbers, found through an if statement with modulo operations. Bubble sorting is applied to the 'evenOnly' array – placing the even integers into the correct order. Finally the original array is run through a for loop; whenever an even integer appears it is replaced by a value in the evenOnly array, which the index is incremented.