Kaitlyn Kottlowski, Sam Law, Justin Gilmour

Professor Sinapova

Problem-Solving Techniques

02/27/20

Problem 01 – CSS (MICS 2018)

Given an integer *n*, print a message of ASCII art that says “CSS”, with the size of the letters scaling to the value of *n*.

i.e. Scaling Factor 1

CCCCC SSSSS SSSSS

C S S

C SSSSS SSSSS

C S S

CCCCC SSSSS SSSSS

Scaling Factor 3

CCCCCCCCCCCCCCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCCCCCCCCCCCCCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCCCCCCCCCCCCCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCC SSS SSS

CCC SSS SSS

CCC SSS SSS

CCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCC SSS SSS

CCC SSS SSS

CCC SSS SSS

CCCCCCCCCCCCCCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCCCCCCCCCCCCCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

CCCCCCCCCCCCCCC SSSSSSSSSSSSSSS SSSSSSSSSSSSSSS

Start by creating an input and having the user enter an integer *n* to represent the number of cases until termination. Create a *scalingFactor* variable to take the remaining inputs from the user to be used for scaling. Create a for loop that runs *n* times. Nested in the for loop, is another for loop that runs for each character in the art dimensions. If it is the first or last set of lines times *scalingFactor* of the art letter “C”, output the character “C” in every space. If it is not the top or bottom set of lines multiplied by *scalingFactor*, output one character “C” followed by *scalingFactor*\*4 characters. Create another for loop outside of the nested loop that will output *scalingFactor*\*5, spaces between the art letters “C” and “S” for each line times *scalingFactor*\*5.

To output the art letter “S”, the process will be similar but not exactly the same. Create a for loop nested inside of the for loop that runs *n* times. If it is the top, middle or bottom sets of lines of the art letter “S” multiplied by scalingFactor, output the character “S” in every space. If it is the second set of lines multiplied by scalingFactor of the art letter “S”, output one character “S” followed by spaces multiplied by scalingFactor\*4. If it is the fourth set of lines multiplied by scalingFactor, output spaces multiplied by scalingFactor\*4 followed by one character “S”.

n = user input(integer)

for i <- cases

scalingFactor = user input(integer)

for j <- scalingFactor \* 5

if j <- scalingFactor OR j >=- scalingFactor \* 4

for k <- scalingFactor \* 5

out = “C”

else

for k <- scalingFactor

out = “C”

for k <- scalingFactor \* 4

out = “ “

for k <- scalingFactor \* 5

out = “ “

if j <- scalingFactor OR j >=- scalingFactor \* 4 OR j >=- scalingFactor \* 2 AND j <- scalingFactor \* 3

for k <- scalingFactor \* 5

out = “S”

else if j >=- scalingFactor AND j <- scalingFactor \* 2

for k <- scalingFactor

out = “S”

for k <- scalingFactor \* 4

out = “ “

else

for k <- scalingFactor \* 4

out = “ “

for k <- scalingFactor

out = “S”

for k <- scalingFactor \* 5

out = “S”

if j <- scalingFactor OR j >=- scalingFactor \* 4 OR j >=- scalingFactor \* 2 AND j <- scalingFactor \* 3

for k <- scalingFactor \* 5

out = “S”

else if j >=- scalingFactor AND j <- scalingFactor \* 2

for k <- scalingFactor

out = “S”

for k <- scalingFactor \* 4

out = “ “

else

for k <- scalingFactor \* 4

out = “ “

for k <- scalingFactor

out = “S”