4.1)

# Input: An array of integers and a positive number

# Output: Returns the integers in the array that are divisible by the given number

def is\_it\_divisible(array, number):

    nums\_divisible = []

    for i in array:

        if i % number == 0:

            # Adds i to the list if its value is divisible by the given number.

            nums\_divisible.append(i)

    return nums\_divisible

print(is\_it\_divisible([20, 21, 25, 28, 33, 34, 35, 36, 41, 42], 7))

print(is\_it\_divisible([18, 54, 76, 81, 36, 48, 99], 9))

Output:

[21, 28, 35, 42]

[18, 54, 81, 36, 99]

4.2)

# Input: An array of real numbers

# Output: Returns the smallest gap between two numbers in the given array

def find\_smallest\_gap(A):

    size\_array = len(A)

    # sets gap equal to difference between the first two numbers in the array.

    gap = abs(A[1] - A[0])

    for i in range(size\_array-1):

        for j in range(i+1, size\_array):

            if abs(A[i]-A[j]) < gap:

                # if the difference is smaller than the previous gap, it becomes the new gap.

                gap = abs(A[i] - A[j])

    return gap

print(find\_smallest\_gap([50, 120, 250, 100, 20, 300, 200]))

print(find\_smallest\_gap([12.4, 45.9, 8.1, 79.8, -13.64, 5.09]))

Output:

20

3.01

4.3)

# Input: A positive integer n and two nxn matrices of numbers.

# Output: Returns the product of the two nxn matrices.

def multiply\_matrices(n, A, B):

    # Creates a matrix of size n, 0 initialized to all indices.

    ans = [[0] \* n for i in range(n)]

    for i in range(n):

        for j in range(n):

            for k in range(n):

                # Multiply the i-th row of A and the j-th column of B and add it to that index.

                ans[i][j] += A[i][k] \* B[k][j]

    return ans

test\_one = multiply\_matrices(2, [[2, 7], [3, 5]], [[8, -4], [6, 6]])

for row in test\_one:

    print(row)

test\_two = multiply\_matrices(3, [[1, 0, 2], [3, -2, 5], [6, 2, -3]], [[0.3, 0.25, 0.1], [0.4, 0.8, 0], [-0.5, 0.75, 0.6]])

for row in test\_two:

    print(row)

Output:

[58, 34]

[54, 18]

[-0.7, 1.75, 1.3]

[-2.4000000000000004, 2.9, 3.3]

[4.1, 0.8500000000000001, -1.1999999999999997]