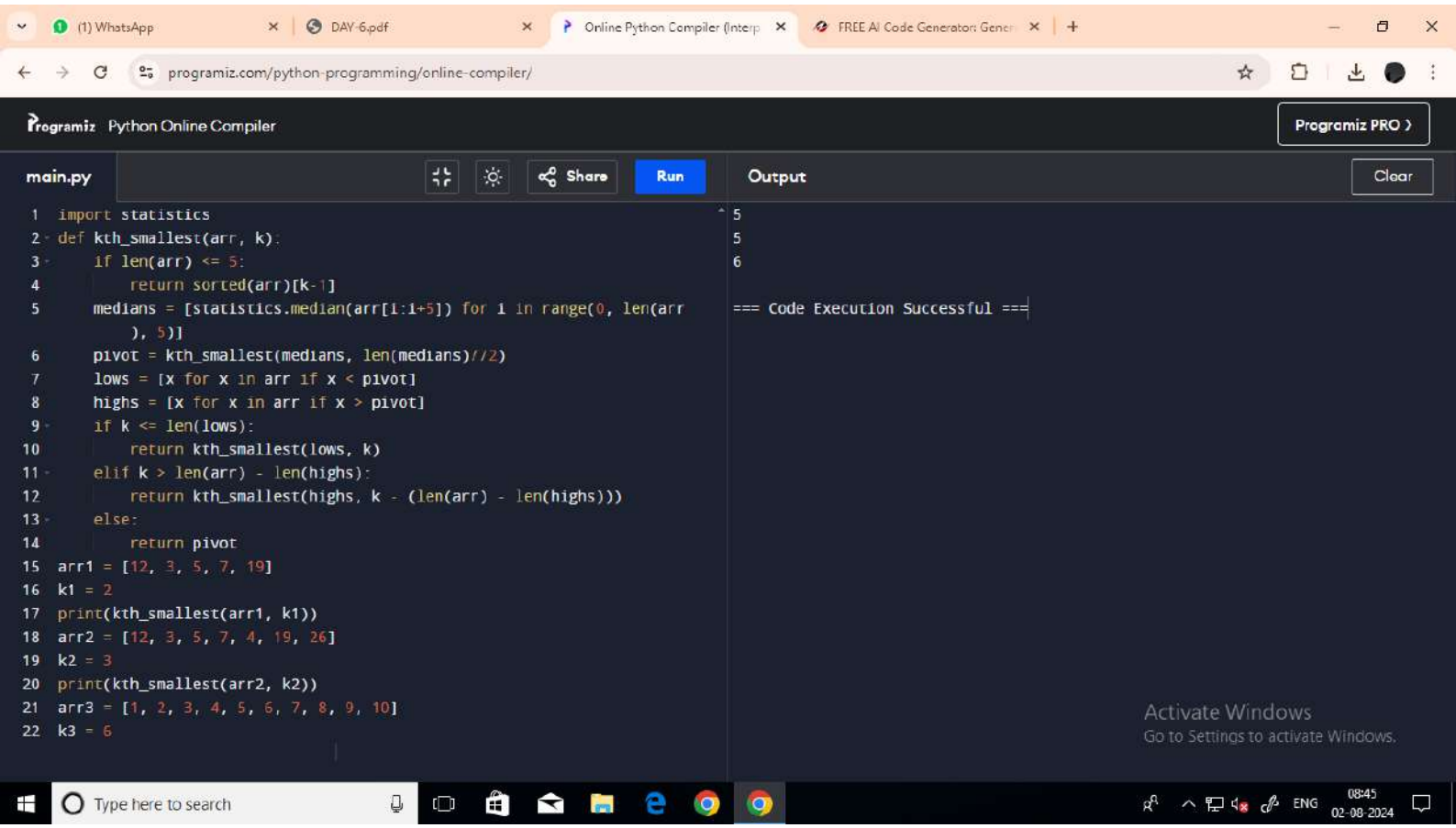
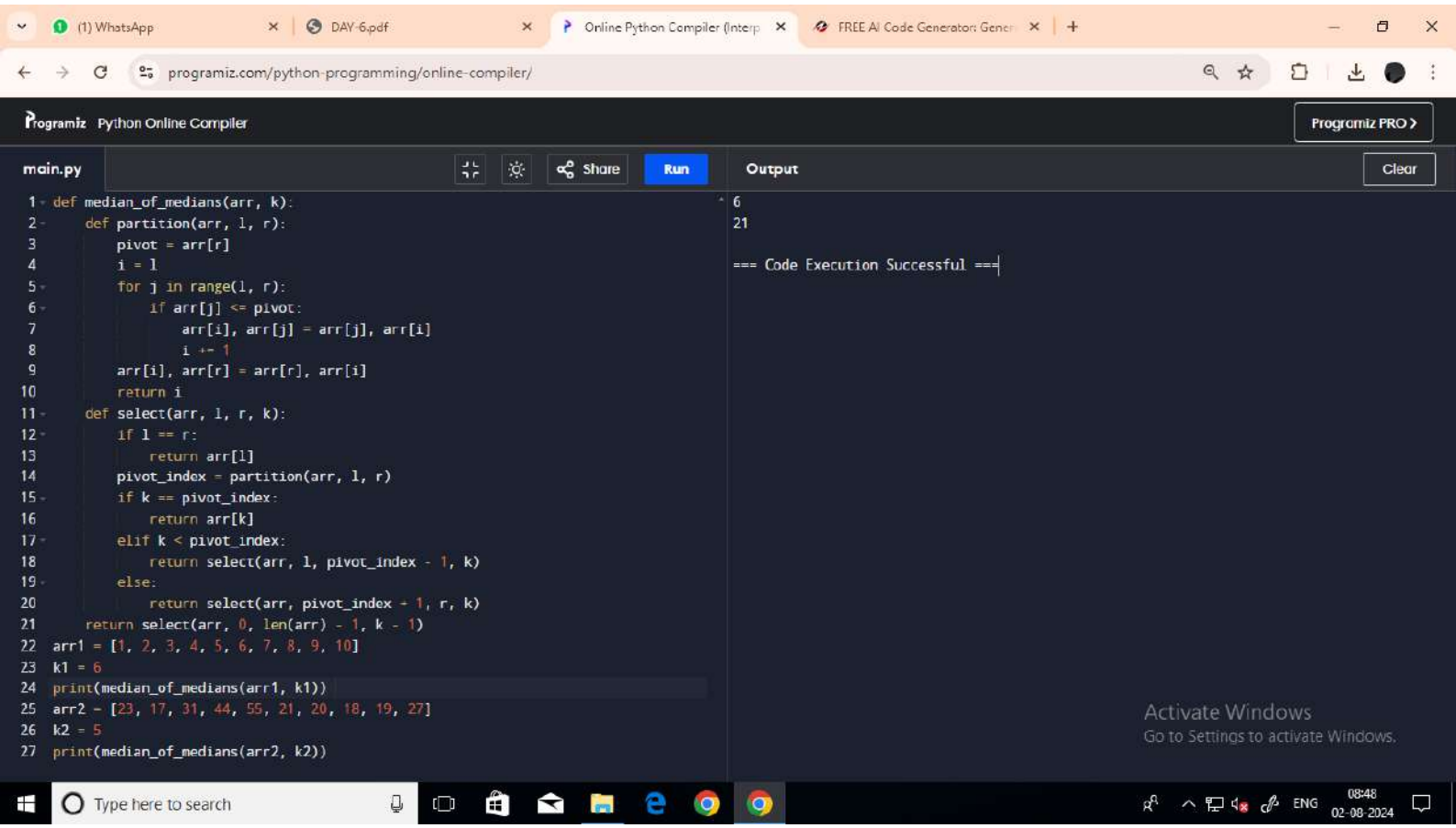
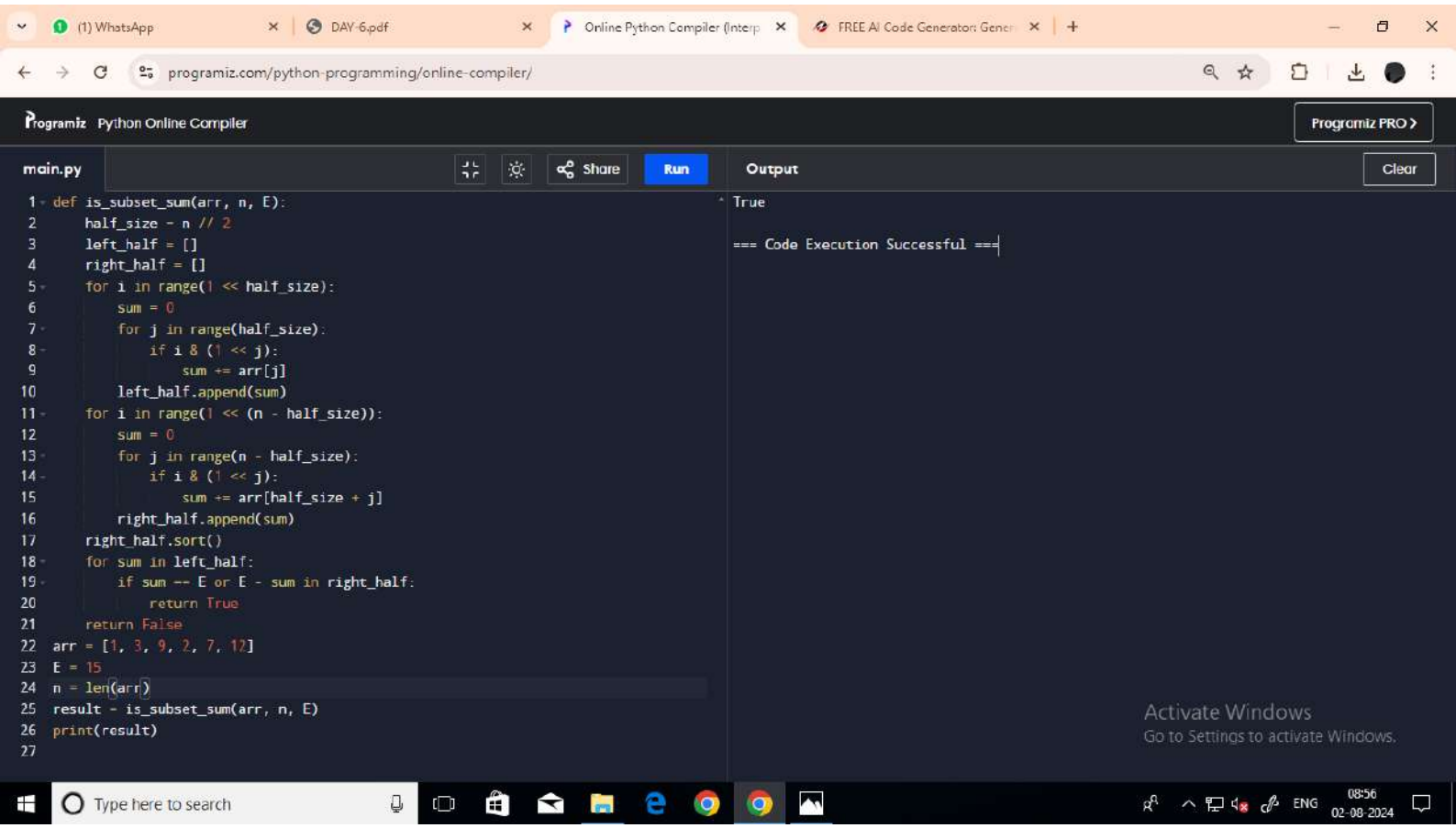
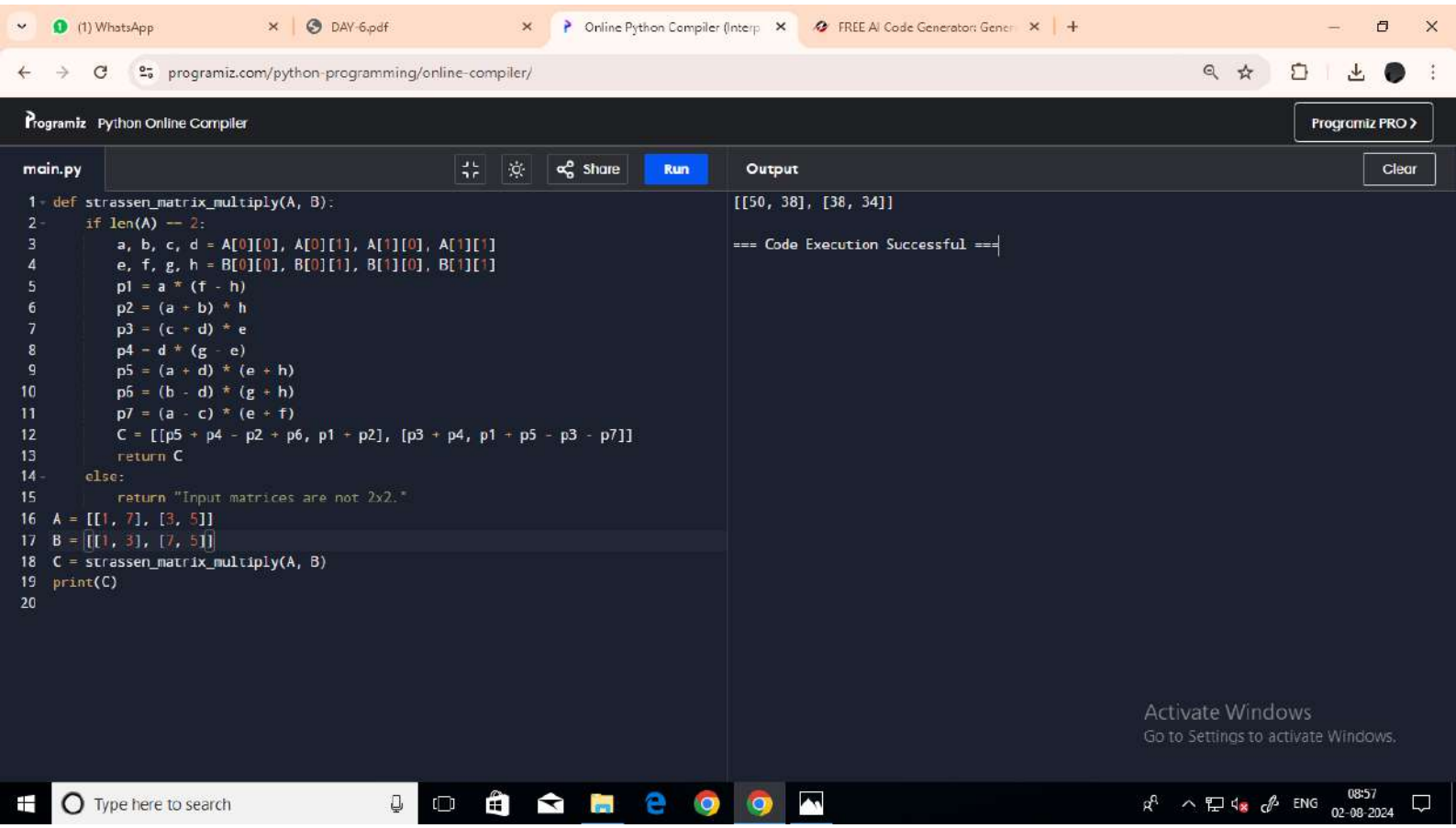

(1) WhatsApp
DAY-6.pdf
Online Python Compiler (Interp
FREE AI Code Generator: Gener
programiz.com/python-programming/online-compiler/
Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
import statistics
def kth_smallest(arr, k):
if len(arr) <= 5:
return sorted(arr)[k-1]
medians = [statistics.median(arr[i:i+5]) for i in range(0, len(arr
), 5)]
pivot = kth_smallest(medians, len(medians)//2)
lows = [x for x in arr if x < pivot]
highs = [x for x in arr if x > pivot]
if k <= len(lows):
return kth_smallest(lows, k)
elif k > len(arr) - len(highs):
return kth_smallest(highs, k - (len(arr) - len(highs)))
else:
return pivot
arr1 = [12, 3, 5, 7, 19]
k1 = 2
print(kth_smallest(arr1, k1))
arr2 = [12, 3, 5, 7, 4, 19, 26]
k2 = 3
print(kth_smallest(arr2, k2))
arr3 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
k3 = 6
5
5
6
=== Code Execution Successful ===
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
08:45
02-08-2024


(1) WhatsApp
DAY-6.pdf
Online Python Compiler (Interp
FREE AI Code Generator: Gener
programiz.com/python-programming/online-compiler/
Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def median_of_medians(arr, k):
def partition(arr, l, r):
pivot = arr[r]
i = l
for j in range(l, r):
if arr[j] <= pivot:
arr[i], arr[j] = arr[j], arr[i]
i += 1
arr[i], arr[r] = arr[r], arr[i]
return i
def select(arr, l, r, k):
if l == r:
return arr[l]
pivot_index = partition(arr, l, r)
if k == pivot_index:
return arr[k]
elif k < pivot_index:
return select(arr, l, pivot_index - 1, k)
else:
return select(arr, pivot_index + 1, r, k)
return select(arr, 0, len(arr) - 1, k - 1)
arr1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
k1 = 6
print(median_of_medians(arr1, k1))
arr2 = [23, 17, 31, 44, 55, 21, 20, 18, 19, 27]
k2 = 5
print(median_of_medians(arr2, k2))
6
21
=== Code Execution Successful ===
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
08:48
02-08-2024


(1) WhatsApp
DAY-6.pdf
Online Python Compiler (Interp
FREE AI Code Generator: Gener
programiz.com/python-programming/online-compiler/
Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def is_subset_sum(arr, n, E):
half_size = n // 2
left_half = []
right_half = []
for i in range(1 << half_size):
sum = 0
for j in range(half_size):
if i & (1 << j):
sum += arr[j]
left_half.append(sum)
for i in range(1 << (n - half_size)):
sum = 0
for j in range(n - half_size):
if i & (1 << j):
sum += arr[half_size + j]
right_half.append(sum)
right_half.sort()
for sum in left_half:
if sum == E or E - sum in right_half:
return True
return False
arr = [1, 3, 9, 2, 7, 12]
E = 15
n = len(arr)
result = is_subset_sum(arr, n, E)
print(result)
True
=== Code Execution Successful ===
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
08:56
02-08-2024


(1) WhatsApp
DAY-6.pdf
Online Python Compiler (Interp
FREE AI Code Generator: Gener
programiz.com/python-programming/online-compiler/
Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def strassen_matrix_multiply(A, B):
if len(A) == 2:
a, b, c, d = A[0][0], A[0][1], A[1][0], A[1][1]
e, f, g, h = B[0][0], B[0][1], B[1][0], B[1][1]
p1 = a * (f - h)
p2 = (a + b) * h
p3 = (c + d) * e
p4 = d * (g - e)
p5 = (a + d) * (e + h)
p6 = (b - d) * (g + h)
p7 = (a - c) * (e + f)
C = [[p5 + p4 - p2 + p6, p1 + p2], [p3 + p4, p1 + p5 - p3 - p7]]
return C
else:
return "Input matrices are not 2x2."
A = [[1, 7], [3, 5]]
B = [[1, 3], [7, 5]]
C = strassen_matrix_multiply(A, B)
print(C)
[[50, 38], [38, 34]]
=== Code Execution Successful ===
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
08:57
02-08-2024

(1) WhatsApp | DAY-6.pdf | Online Python Compiler (Interp | FREE AI Code Generator: Gener

programiz.com/python-programming/online-compiler/

Programiz Python Online Compiler

Programiz PRO >

main.py

Share | Run

```python
def karatsuba(x, y):
    if x < 10 or y < 10:
        return x * y
    m = max(len(str(x)), len(str(y)))
    m2 = m // 2
    high1, low1 = divmod(x, 10**m2)
    high2, low2 = divmod(y, 10**m2)
    z0 = karatsuba(low1, low2)
    z1 = karatsuba((low1 + high1), (low2 + high2))
    z2 = karatsuba(high1, high2)
    return z2 * 10**(2*m2) + (z1 - z2 - z0) * 10**m2 + z0
x = 1234
y = 5678
z = karatsuba(x, y)
print(f"Result of {x} x {y} = {z}")

```

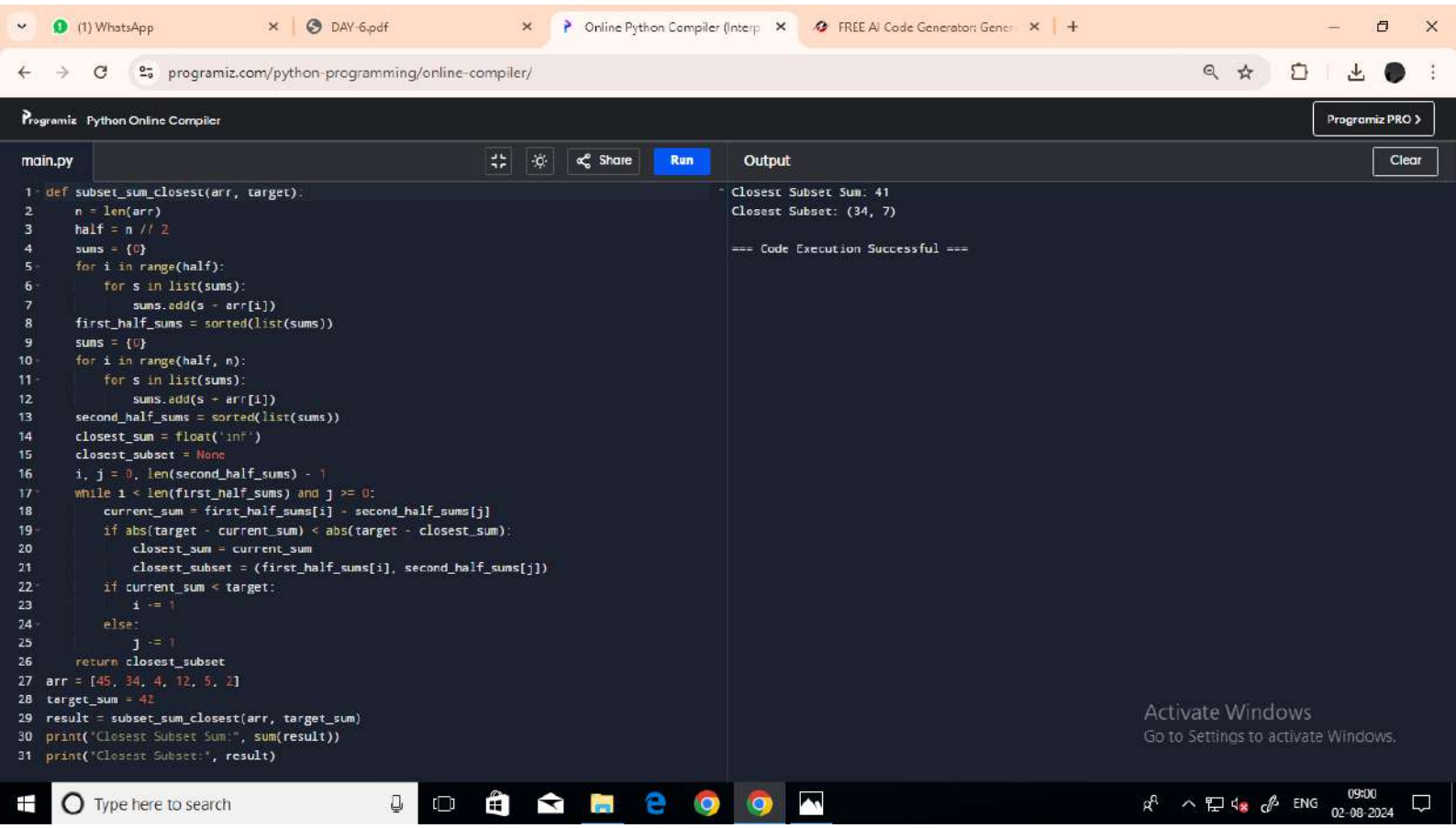
Output | Clear

```
Result of 1234 x 5678 = 7006652

=== Code Execution Successful ===
```


Type here to search

ENG 08:58 02-08-2024


(1) WhatsApp
DAY-6.pdf
Online Python Compiler (Interp
FREE AI Code Generator: Gener
programiz.com/python-programming/online-compiler/
Programiz Python Online Compiler
Programiz PRO
main.py
Share
Run
Output
Clear
def subset_sum_closest(arr, target):
n = len(arr)
half = n // 2
sums = {0}
for i in range(half):
for s in list(sums):
sums.add(s + arr[i])
first_half_sums = sorted(list(sums))
sums = {0}
for i in range(half, n):
for s in list(sums):
sums.add(s + arr[i])
second_half_sums = sorted(list(sums))
closest_sum = float('inf')
closest_subset = None
i, j = 0, len(second_half_sums) - 1
while i < len(first_half_sums) and j >= 0:
current_sum = first_half_sums[i] + second_half_sums[j]
if abs(target - current_sum) < abs(target - closest_sum):
closest_sum = current_sum
closest_subset = (first_half_sums[i], second_half_sums[j])
if current_sum < target:
i += 1
else:
j -= 1
return closest_subset
arr = [45, 34, 4, 12, 5, 2]
target_sum = 42
result = subset_sum_closest(arr, target_sum)
print("Closest Subset Sum:", sum(result))
print("Closest Subset:", result)
Closest Subset Sum: 41
Closest Subset: (34, 7)
=== Code Execution Successful ===
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
09:00
02-08-2024