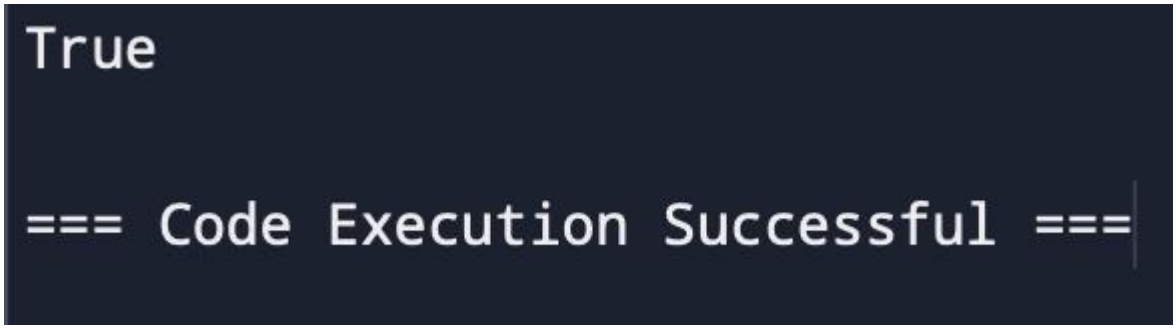
169. Write a program to implement Meet in the Middle Technique. Given a large array of integers and an exact sum E, determine if there is any subset that sums exactly to E. Utilize the Meet in the Middle technique to handle the potentially large size of the array. Return true if there is a subset that sums exactly to E, otherwise return false.

a) E = {1, 3, 9, 2, 7, 12} exact Sum = 15
b) E = {3, 34, 4, 12, 5, 2} exact Sum = 15

PROGRAM :-

```
def is_subset_sum(arr, n, E):
    half = n // 2
    subset1 = []
    subset2 = []

    for i in range(1 << half):
        sum = 0
        for j in range(half):
            if i & (1 << j):
                sum += arr[j]
        subset1.append(sum)

    for i in range(1 << (n - half)):
        sum = 0
        for j in range(n - half):
            if i & (1 << j):
                sum += arr[half + j]
        subset2.append(sum)

    subset1.sort()

    for s in subset2:
        if s == E or E - s in subset1:
            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)
```

OUTPUT:-



TIME COMPLEXITY:- O(2n/2)