87. Meet in middle technique

Code:

```
def subset_sum(nums, target):
    from itertools import chain, combinations

    def all_sums(nums):
        """Generate all possible sums of subsets of nums."""
        return {sum(combo) for combo in chain(*[combinations(nums, r) for r in range(len(nums) + 1)])}


    mid = len(nums) // 2
    left, right = nums[:mid], nums[mid:]
    left_sums = all_sums(left)
    right_sums = all_sums(right)
    for l_sum in left_sums:
        if (target - l_sum) in right_sums:
            return True

    return False
nums = [3, 34, 4, 12, 5, 2]
target = 9
print(subset_sum(nums, target))
```

Output:

```
True
```

Time Complexity:

- T(n)= O(2^n)