```python
def binarySearch (arr, l, r, x):
    if r >= l:
        mid = l + (r - l) // 2
        if arr[mid] == x:
            return mid
        elif arr[mid] > x:
            return binarySearch(arr, mid+1, r, x)
```
Talent Battle 100 Days Coding Series
```python
        else:
            return binarySearch(arr, l, mid-1, x)
    else:
        return -1
for _ in range(int(input().strip())):
    n = int(input().strip())
    sub_sums_list = list(map(int, input().strip().split()))
    sub_sums = sorted(sub_sums_list, reverse = True)
    sub_sums.pop()
    original_set = []
    to_be_removed = []

    while len(original_set) < n:
        element = sub_sums.pop()
        original_set.append(element)
        will_be_removed = [element]
        for rem_val in to_be_removed:
            new_rem_val = rem_val + element
            will_be_removed.append(new_rem_val)
            idx = binarySearch(sub_sums, 0, len(sub_sums) - 1, new_rem_val)
            sub_sums.pop(idx)
        to_be_removed += will_be_removed

    print(*sorted(original_set),sep = " ")
```