```
# Task 1
True
True
False
Final roster: ['A', 'B']
# Task 2
False
is_full: True
# Task 3
True
False
Roster now: ['Alex']
# Task 4
Seats left before: 1
Seats left after fill: 0
Roster: ['Alex', 'Priya', 'Jordan']
```

```
# Task 5
Added count: 2
Final roster: ['A', 'B']

# Task 6
Drop Jordan (1): True
Roster: ['Lee']
Drop Jordan (2): False
Roster: ['Lee']

# Task 7
Transfer when dst full: False
alg roster: ['Priya', 'Alex']
ds roster: ['Sam']
Transfer when dst not full: True
alg roster: ['Priya']
ds roster: ['Alex']
```

```
# Task 8
Found EECE2560: EECE2560 (Algorithms): <empty>
Missing EECE9999: None

# Task 9
is_full before: True
is_full after: False
Add one more: True
Roster: ['A', 'B', 'C']

# Task 10
Caught: Capacity cannot be negative

# Task 11
EECE3000 (EmptyCourse): <empty>
EECE3000 (EmptyCourse): Alex, Priya
```

```
# Task 12
Merged cap: 3
Merged roster: ['Alex', 'Priya', 'Zara']
Num dropped: 0

# Task 13
Before: [] | wait: ['A', 'B', 'C', 'D']
Added: 3
After: ['A', 'B', 'C'] | wait: ['D']
Before: ['I'] | wait: ['J', 'K']
Added: 1
After: ['I', 'J'] | wait: ['K']

# Task 14
Returning booleans/counts avoids aliasing confusion.
```