```python
print("Q.1")
from collections import defaultdict
enrollments = [
    ('A', {'CS101', 'MATH101'}),
    ('B', {'CS101', 'PHYS101'}),
    ('C', {'CS101', 'MATH101', 'PHYS101'}),
    ('D', {'MATH101', 'PHYS101'}),
    ('E', {'CS101', 'MATH101'}),
]
course_combinations = defaultdict(list)

for student, courses in enrollments:
    if len(courses) > 1:
        frozen_courses = frozenset(courses)
        course_combinations[frozen_courses].append(student)


for course_combination, students in course_combinations.items():
    print(f"Courses: {course_combination}, Students: {students}")

print()
print("Q.2")

combination_count = {combination: len(students) for combination, students in course_combinations.items()}
for combination, count in combination_count.items():
    print(f"Courses: {combination}, Number of Students: {count}")

print()
print("Q.3")
top_3_combinations = sorted(combination_count.items(), key=lambda x: x[1], reverse=True)[:3]
print("Top 3 most popular course combinations:")
for combination, count in top_3_combinations:
    print(f"Courses: {combination}, Number of Students: {count}")
```

```
Q.1
Courses: frozenset({'MATH101', 'CS101'}), Students: ['A', 'E']
Courses: frozenset({'PHYS101', 'CS101'}), Students: ['B']
Courses: frozenset({'MATH101', 'PHYS101', 'CS101'}), Students: ['C']
Courses: frozenset({'MATH101', 'PHYS101'}), Students: ['D']

Q.2
Courses: frozenset({'MATH101', 'CS101'}), Number of Students: 2
Courses: frozenset({'PHYS101', 'CS101'}), Number of Students: 1
Courses: frozenset({'MATH101', 'PHYS101', 'CS101'}), Number of Students: 1
Courses: frozenset({'MATH101', 'PHYS101'}), Number of Students: 1

Q.3
Top 3 most popular course combinations:
Courses: frozenset({'MATH101', 'CS101'}), Number of Students: 2
Courses: frozenset({'PHYS101', 'CS101'}), Number of Students: 1
Courses: frozenset({'MATH101', 'PHYS101', 'CS101'}), Number of Students: 1
```