

```

# Each student will be represented as a tuple: (name, roll_no, (mark1,
mark2, mark3), grade)
students = []

# 1. Create Students
def create_students():
    global students
    students = [
        ("Alice", 101, (85, 90, 78), "A"),
        ("Bob", 102, (75, 80, 70), "B"),
        ("Charlie", 103, (95, 88, 92), "A+")
    ]

# 2. Display All Students
def display_all_students():
    for student in students:
        print(f"Name: {student[0]}, Roll No: {student[1]}, Marks:
{student[2]}, Grade: {student[3]}")

# 3. Add a New Student
def add_student(name, roll_no, marks, grade):
    students.append((name, roll_no, marks, grade))
    print(f"Student {name} added successfully!")

# 4. Search for a Student
def search_student(roll_no):
    for student in students:
        if student[1] == roll_no:
            print(f"Student Found: Name: {student[0]}, Roll No:
{student[1]}, Marks: {student[2]}, Grade: {student[3]}")
            return
    print("Student not found.")

# 5. Calculate Total Marks
def calculate_total_marks():
    for student in students:
        total = sum(student[2])
        print(f"Name: {student[0]}, Roll No: {student[1]}, Total
Marks: {total}")

# 6. Update Grades
def update_grade(roll_no, new_grade):
    for i, student in enumerate(students):
        if student[1] == roll_no:
            updated_student = (student[0], student[1], student[2],
new_grade)
            students[i] = updated_student
            print(f"Grade updated for {student[0]}")
            return
    print("Student not found.")

```

```

# 7. Remove a Student
def remove_student(roll_no):
    global students
    students = [student for student in students if student[1] !=
roll_no]
    print(f"Student with Roll No {roll_no} removed, if existed.")

# Example usage
create_students()
print("\nAll Students:")
display_all_students()

print("\nAdding a New Student:")
add_student("David", 104, (88, 77, 69), "B+")

print("\nAll Students After Adding:")
display_all_students()

print("\nSearch for Student Roll No 102:")
search_student(102)

print("\nTotal Marks of Each Student:")
calculate_total_marks()

print("\nUpdate Grade for Roll No 101:")
update_grade(101, "A+")

print("\nAll Students After Grade Update:")
display_all_students()

print("\nRemove Student with Roll No 103:")
remove_student(103)

print("\nAll Students After Removal:")
display_all_students()

```

All Students:

Name: Alice, Roll No: 101, Marks: (85, 90, 78), Grade: A

Name: Bob, Roll No: 102, Marks: (75, 80, 70), Grade: B

Name: Charlie, Roll No: 103, Marks: (95, 88, 92), Grade: A+

Adding a New Student:

Student David added successfully!

All Students After Adding:

Name: Alice, Roll No: 101, Marks: (85, 90, 78), Grade: A

Name: Bob, Roll No: 102, Marks: (75, 80, 70), Grade: B

Name: Charlie, Roll No: 103, Marks: (95, 88, 92), Grade: A+

Name: David, Roll No: 104, Marks: (88, 77, 69), Grade: B+

Search for Student Roll No 102:

Student Found: Name: Bob, Roll No: 102, Marks: (75, 80, 70), Grade: B

Total Marks of Each Student:

Name: Alice, Roll No: 101, Total Marks: 253

Name: Bob, Roll No: 102, Total Marks: 225

Name: Charlie, Roll No: 103, Total Marks: 275

Name: David, Roll No: 104, Total Marks: 234

Update Grade for Roll No 101:

Grade updated for Alice

All Students After Grade Update:

Name: Alice, Roll No: 101, Marks: (85, 90, 78), Grade: A+

Name: Bob, Roll No: 102, Marks: (75, 80, 70), Grade: B

Name: Charlie, Roll No: 103, Marks: (95, 88, 92), Grade: A+

Name: David, Roll No: 104, Marks: (88, 77, 69), Grade: B+

Remove Student with Roll No 103:

Student with Roll No 103 removed, if existed.

All Students After Removal:

Name: Alice, Roll No: 101, Marks: (85, 90, 78), Grade: A+

Name: Bob, Roll No: 102, Marks: (75, 80, 70), Grade: B

Name: David, Roll No: 104, Marks: (88, 77, 69), Grade: B+