

EX.No:2	Analysis of student scores to determine statistical measures and grade classification.
DATE:	

AIM:

To write a Python program to analyze student scores, calculates statistical measures and classifies each student grades based on score ranges.

ALGORITHM:

- Step 1: Start
- Step 2: Accept scores of n students into a list.
- Step 3: Calculate

Mean: Sum of all scores divided by number of scores.

Median:

- Sort the list manually (e.g., using bubble sort).
- If the number of scores is odd, pick the middle one.
- If even, average the two middle scores.

Maximum and Minimum: Traverse the list and compare each value.

Standard Deviation:

- Compute mean first.
- Then compute variance as average of squared differences from the mean.
- Take square root of variance (use exponent `**0.5`).

- Step 4: Assign grades:

A: 90–100

B: 80–89

C: 70–79

D: 60–69

F: Below 60

- Step 5: Display all statistical values and grade-wise classification.
- Step 6: End

PROGRAM:

Step 1: Input scores

```
scores = list(map(int, input("Enter scores separated by space: ").split()))
```

```
n = len(scores)
```

```
# Step 2: Mean
```

```
total = 0
```

```
for score in scores:
```

```
    total += score
```

```
mean = total / n
```

```
# Step 3: Sorting (bubble sort for median)
```

```
for i in range(n):
```

```
    for j in range(0, n - i - 1):
```

```
        if scores[j] > scores[j + 1]:
```

```
            scores[j], scores[j + 1] = scores[j + 1], scores[j]
```

```
# Step 4: Median
```

```
if n % 2 == 0:
```

```
    median = (scores[n // 2 - 1] + scores[n // 2]) / 2
```

```
else:
```

```
    median = scores[n // 2]
```

```
# Step 5: Maximum and Minimum
```

```
maximum = scores[0]
```

```
minimum = scores[0]
```

```
for score in scores:
```

```
    if score > maximum:
```

```
        maximum = score
```

```
    if score < minimum:
```

```
        minimum = score
```

```
# Step 6: Standard Deviation
```

```
sum_sq_diff = 0
for score in scores:
    sum_sq_diff += (score - mean) ** 2
variance = sum_sq_diff / n
std_dev = variance ** 0.5
```

Step 7: Grade Classification

```
def assign_grade(score):
    if score >= 90:
        return 'A'
    elif score >= 80:
        return 'B'
    elif score >= 70:
        return 'C'
    elif score >= 60:
        return 'D'
    else:
        return 'F'

grades = [assign_grade(score) for score in scores]
```

Step 8: Display Results

```
print("\n--- Statistical Results ---")
print(f"Mean: {mean:.2f}")
print(f"Median: {median}")
print(f"Maximum: {maximum}")
print(f"Minimum: {minimum}")
print(f"Standard Deviation: {std_dev:.2f}")
print("\n--- Grade Classification ---")
for i in range(n):
    print(f"Student {i+1}: Score = {scores[i]}, Grade = {grades[i]}")
```

Result:

Hence student scores were analysed through statistical measures and grade classification was performed.

EXERCISE:

1. Write a Python program to count and display how many students received each grade.
2. Write a python program to display the names of students (input names separately) with their scores and grades.
3. Modify the code to classify fail/pass ($\text{pass} \geq 40$).
4. Print all students who scored above the average.
5. Write a Python program to compute the Mode of a List of Scores using a built in function.