SRINIVASA EDUCATIONAL SOCIETY'S

PACE INSTITUTE OF TECHNOLOGY & SCIENCES

(AUTONOMOUS)

Approved by AICTE, UGC, New Delhi & Govt. of Andhra Pradesh | Permanently Affiliated to JNTUK, Kakinada, A.P.

ACCREDITED BY **NAAC** WITH **'A'** GRADE | ACCREDITED BY **NBA**

An ISO 9001 : 2008 Certified Institution |'A' Grade Engineering College by Government of A.P.

NH-16, Near Valluramma Temple, ONGOLE - 523 272, A.P., Contact No.: 08592 278315, 9581456310 | www.pace.ac.in

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
(2023-2024)

Internship and mini project based on python programming with Data Engineer

Project title: scholastic achievement dashboard

In accordance with requirement of degree of
BACHELOR OF TECHNOLOGY
In
ELECTRICAL AND ELECTRONICS ENGINEERING

Submitted by:
B.SNEHA
21KQ1A0206

Date: 11-06-2024

STUDENT’S DECLARATION

I ,**G.SNEHA** a student of **Bachelor of Technology** Program, Reg. No. **21KQ1A0206** of the Department of **Electrical and Electronics Engineering, PACE Institute of Technology andSciences, Ongole** do hereby declare that I have completed the INTERNSHIP AND MINI PROJECT BASED ON PYTHON PROGRAMMING WITH DATA ENGINEER, Department of **Electrical and Electronics Engineering**.

(Signature of student & date)

OFFICIAL CERTIFICATION

This is to certify that **G.SNEHA** Reg. No. **21KQ1A0206** has completed his/her INTERNSHIP AND MINI PROJECT BASED ON PYTHON PROGRAMMING WITH DATA ENGINEER under my supervision as a part of partial fulfillment of the requirement for the Degree of **Bachelor of Technology** in the **Electrical and Electronics Engineering**

This report is accepted for evaluation.

Faculty Mentor **Head of the Department**

Scholastic achievement dashboard

Description:- Firstly, it computes the total marks for each student by summing up their scores across all subjects. Secondly, it identifies the minimum and maximum marks attained by each student, providing insights into their performance spectrum. Furthermore, the code sorts the marks of each subject, allowing for a clearer understanding of the distribution and trends within individual subjects. Lastly, it ensures accuracy by avoiding the duplication of marks while calculating the total sum for each student, thereby presenting a reliable overview of their scholastic achievements. By integrating these functionalities, the code offers a comprehensive and organized platform for assessing student performance, aiding educators, students, and parents alike in monitoring progress and identifying areas for improvement.

Requirement:-

1.sum of all subjects score of each student
2.sorting of total score
3.printing the sum of all subjects of each student without duplicates.
4.minmum,maximum marks among all subjects of each student.
5.average marks among all subjects of each student
6.sorting of each student marks in all subjects
7.maximum score in each subject
8.minimum,maximum of sum of total marks in all subjects.

Approach:-

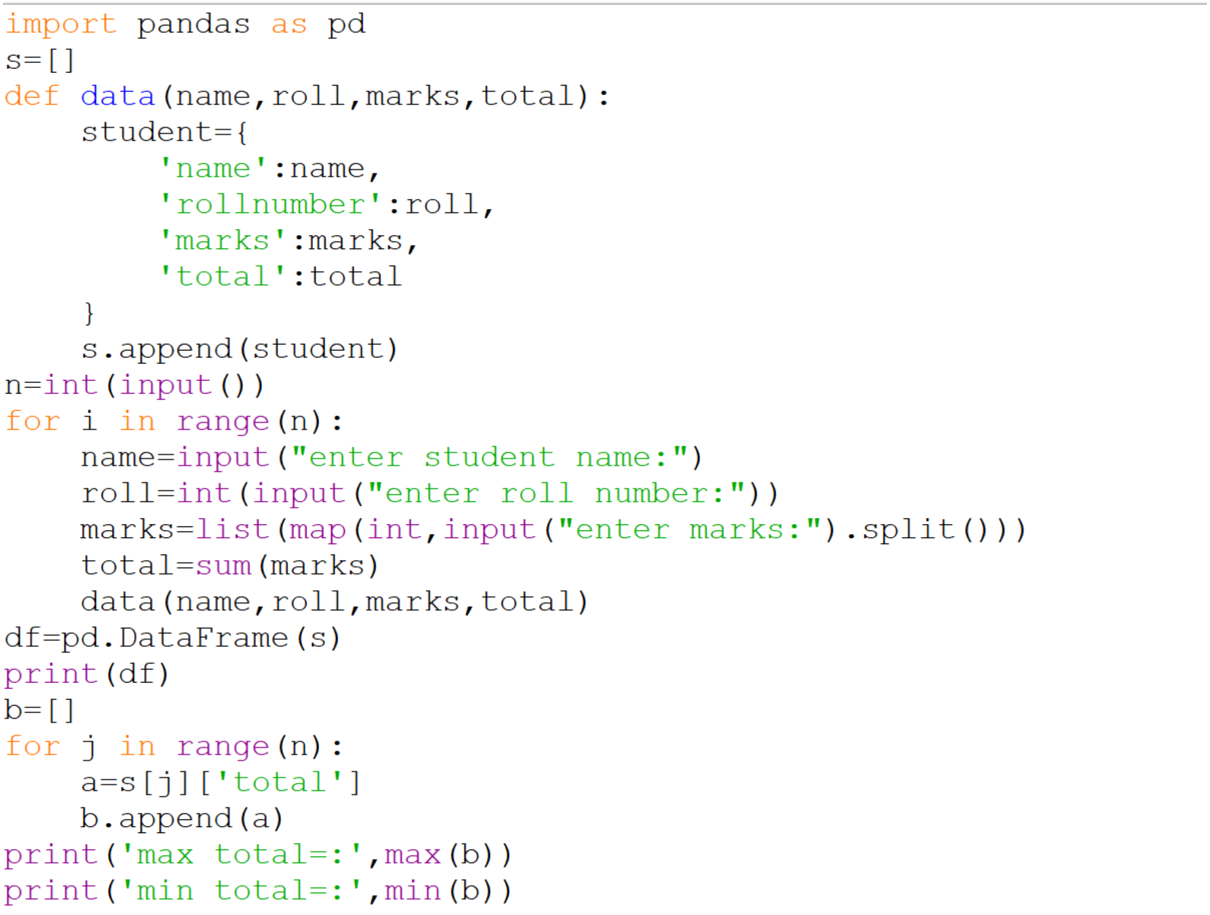
1. Finding the sum of all subjects:
I can create a function that takes in the marks of each student as input and calculates the sum of all subjects for that student. Within this function, you'll iterate over the marks for each subject and accumulate the sum.

2. Minimum and maximum marks gained by each student:
Similar to finding the sum, We can create a function that takes in the marks of each student as input and calculates both the minimum and maximum marks. Within this function, I iterate over the marks and update variables to track the minimum and maximum values.

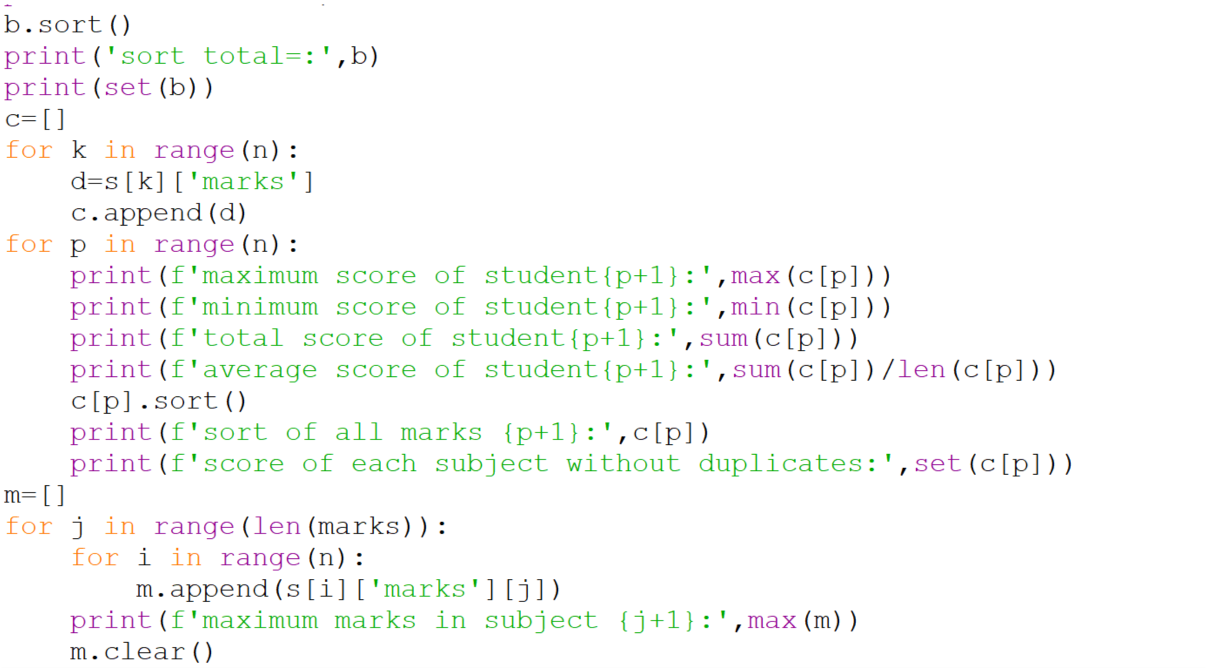
3. Sorting of each subject's marks:
For this functionality, We can create a function that takes in the marks of all students for a particular subject and sorts them in ascending or descending order using Python's built-in sorting functions (sorted() or .sort()).

4. Avoiding duplicate sum of marks:
To prevent counting the same subject's marks multiple times for a student, I can use a data structure like a set or dictionary to keep track of which subjects have been included in the total sum calculation for each student. Before adding marks to the total sum, you'll check if the subject has already been included.

Code:-

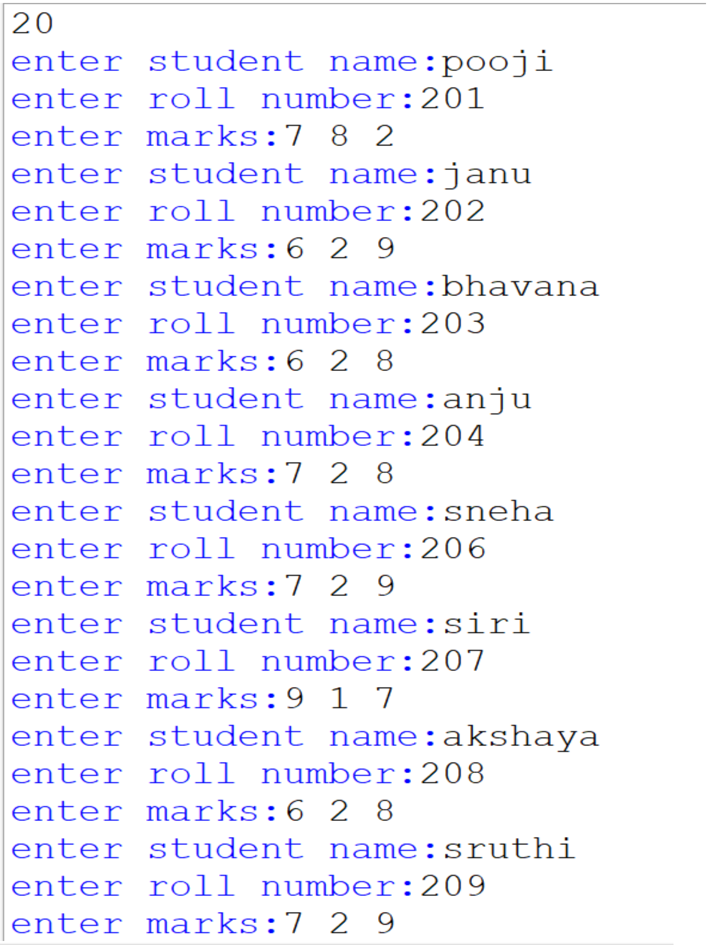


```
import pandas as pd
s=[]
def data(name,roll,marks,total):
    student={
        'name':name,
        'rollnumber':roll,
        'marks':marks,
        'total':total
    }
    s.append(student)
n=int(input())
for i in range(n):
    name=input("enter student name:")
    roll=int(input("enter roll number:"))
    marks=list(map(int,input("enter marks:").split()))
    total=sum(marks)
    data(name,roll,marks,total)
df=pd.DataFrame(s)
print(df)
b=[]
for j in range(n):
    a=s[j]['total']
    b.append(a)
print('max total=:',max(b))
print('min total=:',min(b))
```

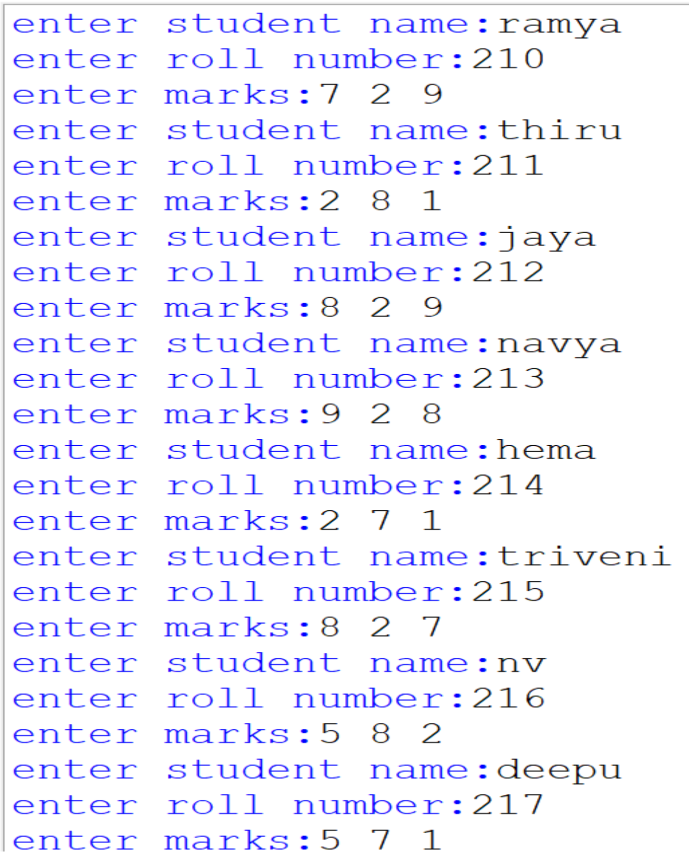


```
b.sort()
print('sort total=:',b)
print(set(b))
c=[]
for k in range(n):
    d=s[k]['marks']
    c.append(d)
for p in range(n):
    print(f'maximum score of student{p+1}:',max(c[p]))
    print(f'minimum score of student{p+1}:',min(c[p]))
    print(f'total score of student{p+1}:',sum(c[p]))
    print(f'average score of student{p+1}:',sum(c[p])/len(c[p]))
    c[p].sort()
    print(f'sort of all marks {p+1}:',c[p])
    print(f'score of each subject without duplicates:',set(c[p]))
m=[]
for j in range(len(marks)):
    for i in range(n):
        m.append(s[i]['marks'][j])
    print(f'maximum marks in subject {j+1}:',max(m))
    m.clear()
```

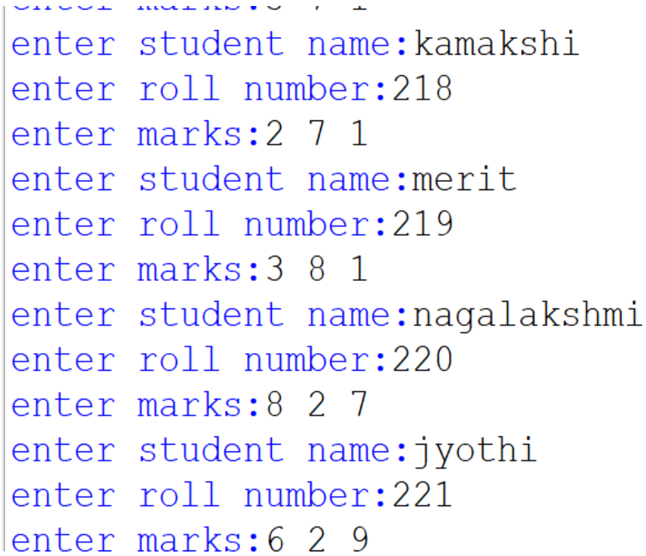
Output:-



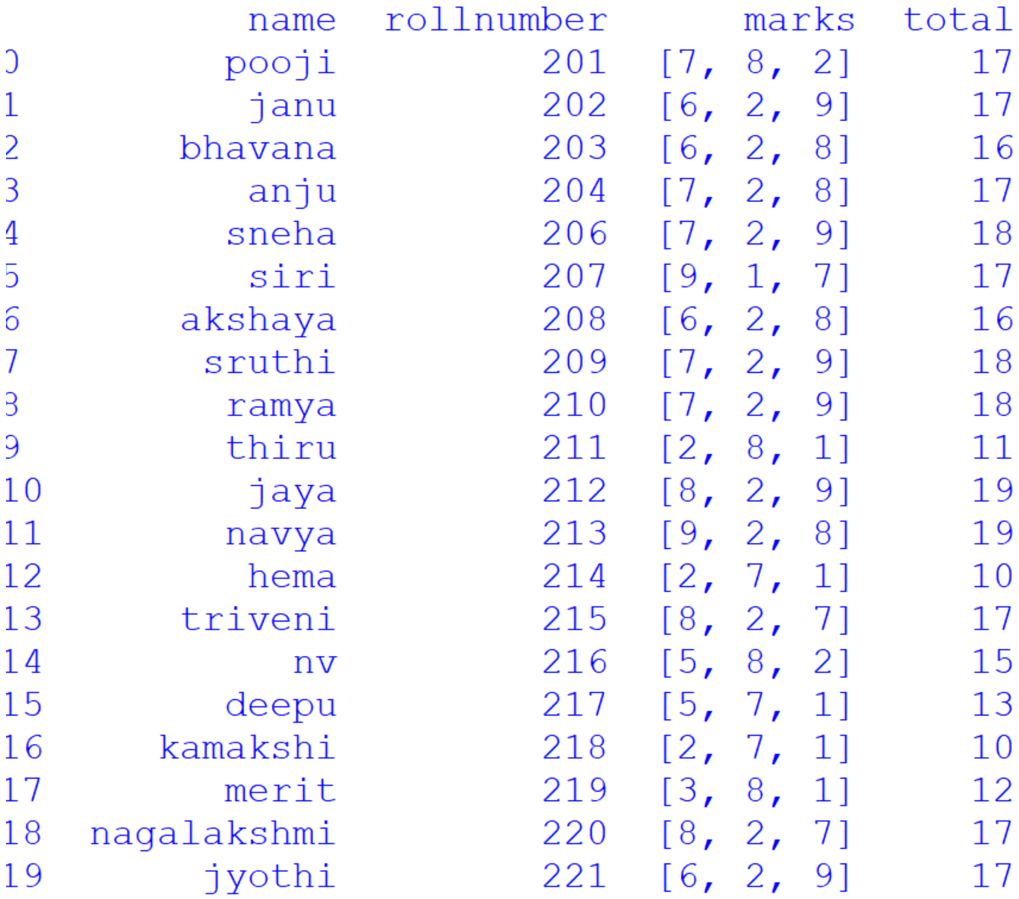
```
20
enter student name:pooji
enter roll number:201
enter marks:7 8 2
enter student name:janu
enter roll number:202
enter marks:6 2 9
enter student name:bhavana
enter roll number:203
enter marks:6 2 8
enter student name:anju
enter roll number:204
enter marks:7 2 8
enter student name:sneha
enter roll number:206
enter marks:7 2 9
enter student name:siri
enter roll number:207
enter marks:9 1 7
enter student name:akshaya
enter roll number:208
enter marks:6 2 8
enter student name:sruthi
enter roll number:209
enter marks:7 2 9
```



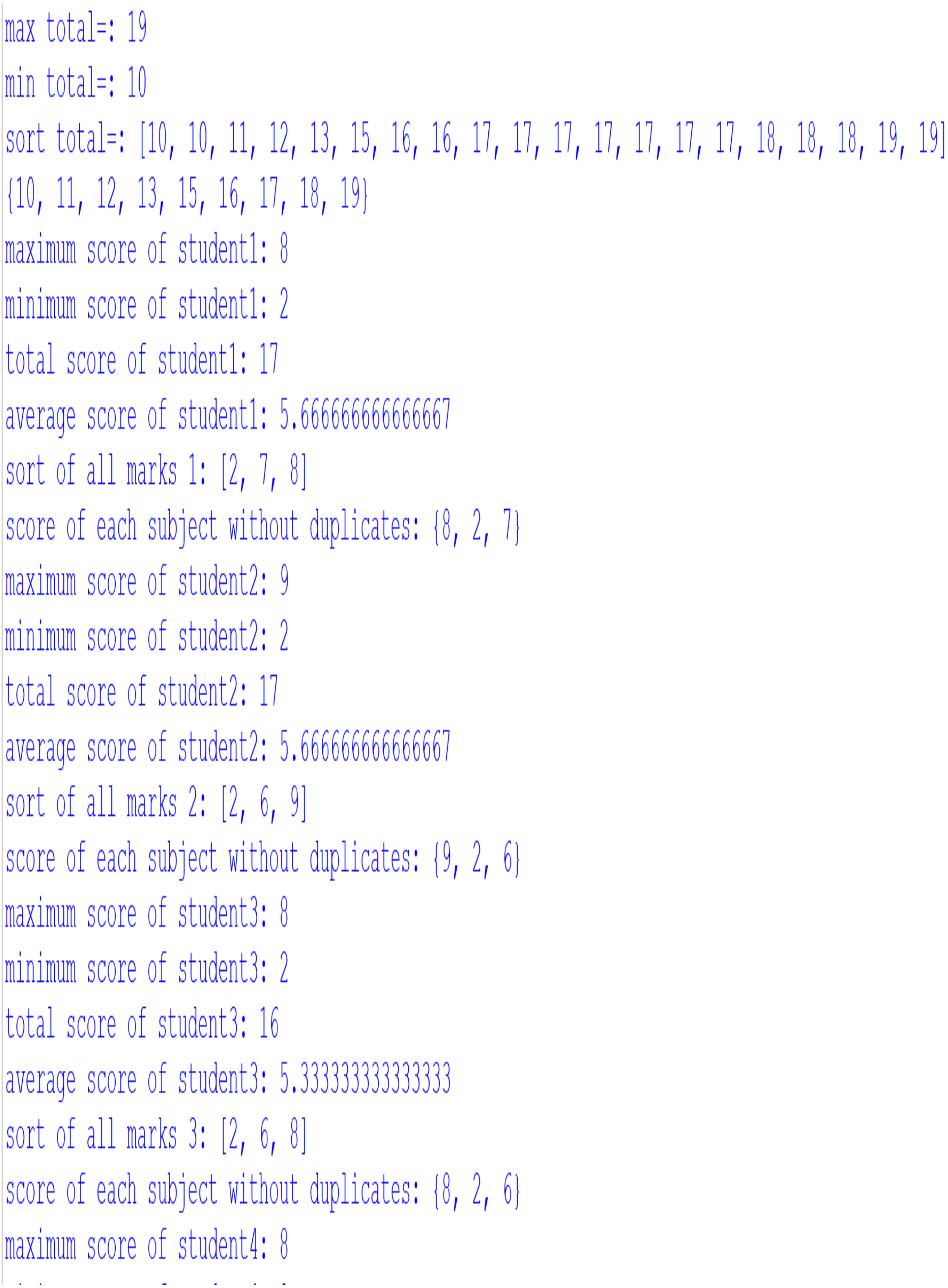
```
enter student name:ramya
enter roll number:210
enter marks:7 2 9
enter student name:thiru
enter roll number:211
enter marks:2 8 1
enter student name:jaya
enter roll number:212
enter marks:8 2 9
enter student name:navya
enter roll number:213
enter marks:9 2 8
enter student name:hema
enter roll number:214
enter marks:2 7 1
enter student name:triveni
enter roll number:215
enter marks:8 2 7
enter student name:nv
enter roll number:216
enter marks:5 8 2
enter student name:deepu
enter roll number:217
enter marks:5 7 1
```



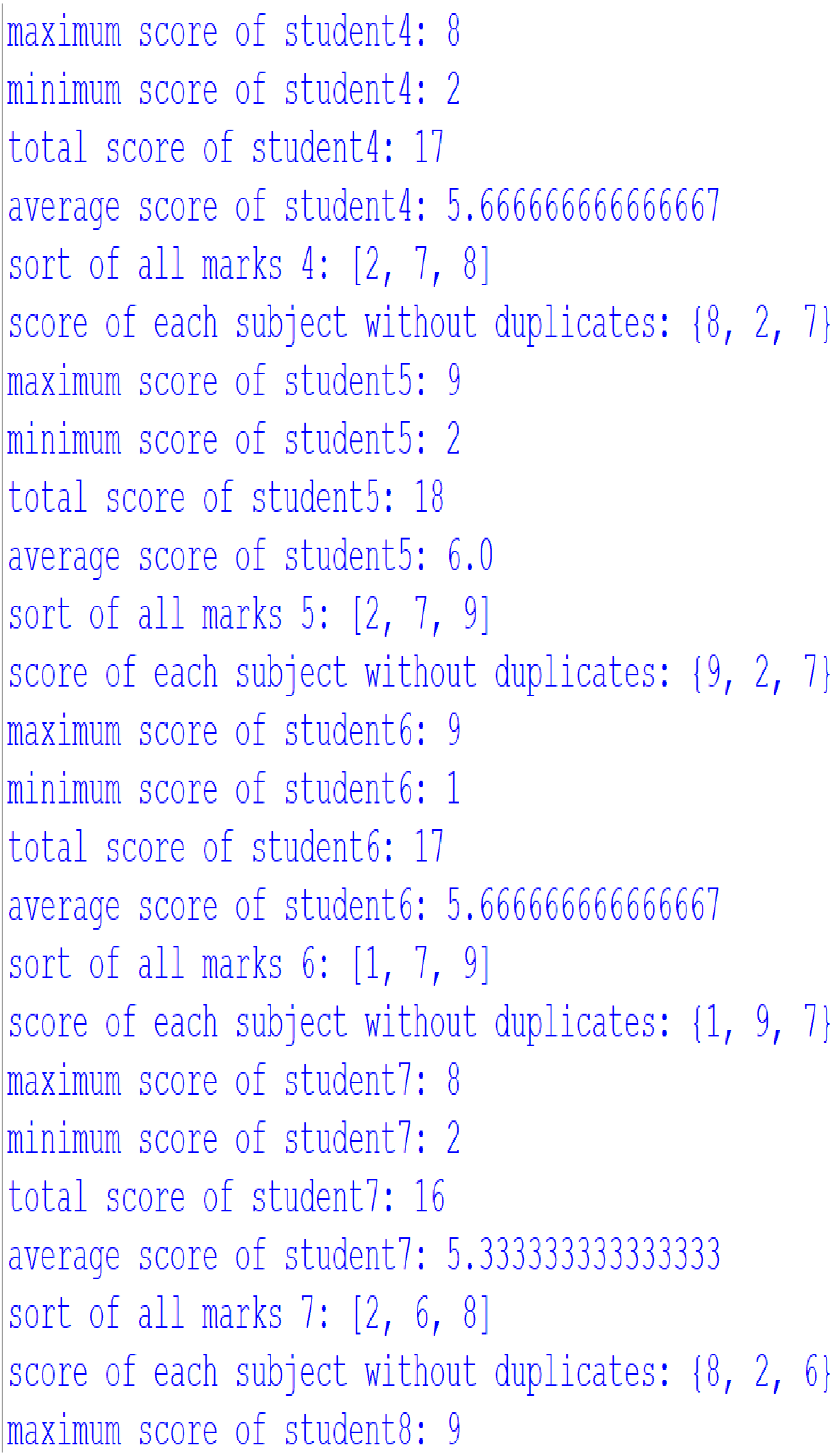
```
enter marks:5 7 1
enter student name:kamakshi
enter roll number:218
enter marks:2 7 1
enter student name:merit
enter roll number:219
enter marks:3 8 1
enter student name:nagalakshmi
enter roll number:220
enter marks:8 2 7
enter student name:jyothi
enter roll number:221
enter marks:6 2 9
```



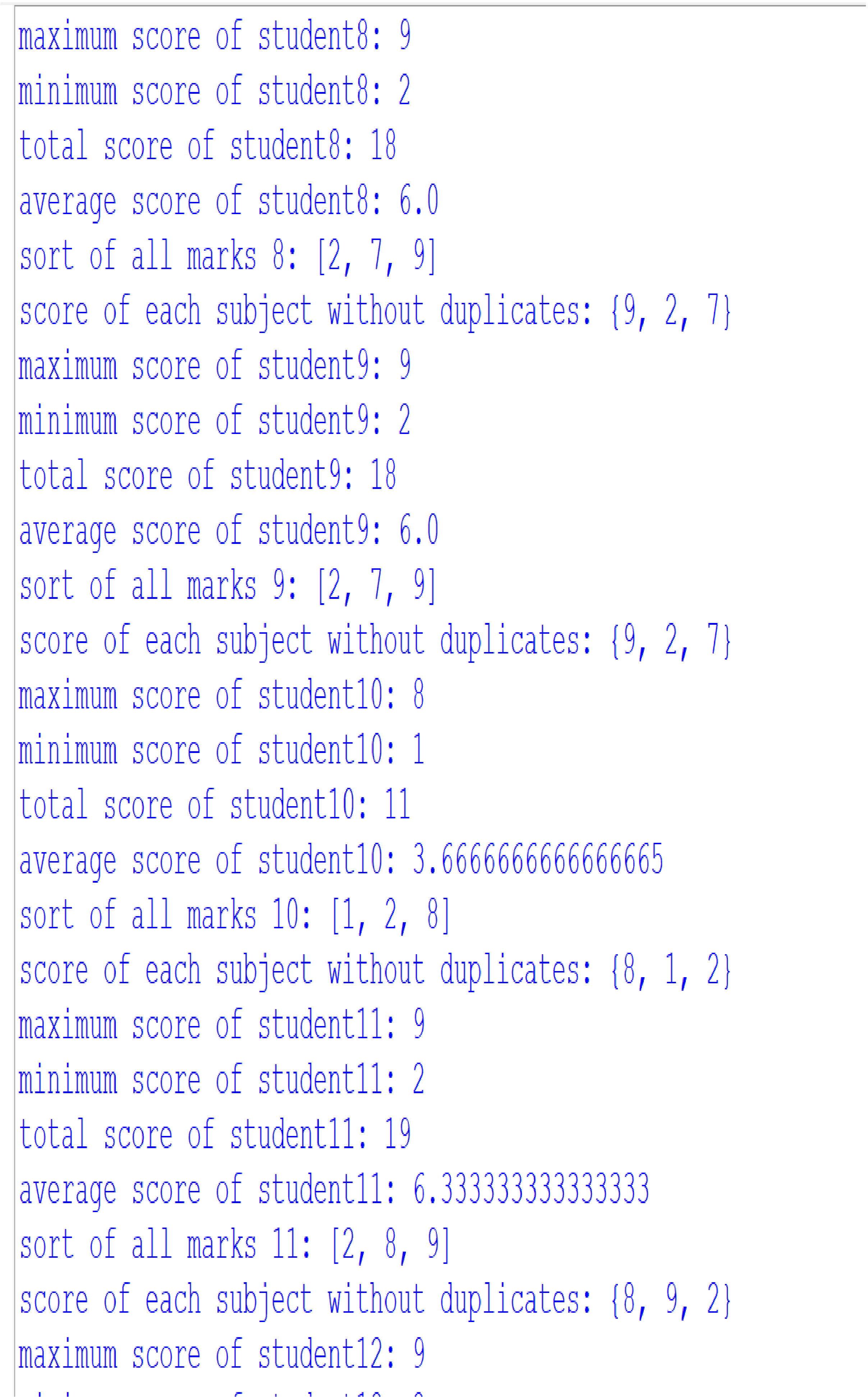
```
           name  rollnumber      marks  total
0         pooji         201  [7, 8, 2]     17
1          janu         202  [6, 2, 9]     17
2       bhavana         203  [6, 2, 8]     16
3          anju         204  [7, 2, 8]     17
4         sneha         206  [7, 2, 9]     18
5          siri         207  [9, 1, 7]     17
6      akshaya         208  [6, 2, 8]     16
7        sruthi         209  [7, 2, 9]     18
8         ramya         210  [7, 2, 9]     18
9         thiru         211  [2, 8, 1]     11
10         jaya         212  [8, 2, 9]     19
11        navya         213  [9, 2, 8]     19
12         hema         214  [2, 7, 1]     10
13      triveni         215  [8, 2, 7]     17
14           nv         216  [5, 8, 2]     15
15        deepu         217  [5, 7, 1]     13
16     kamakshi         218  [2, 7, 1]     10
17        merit         219  [3, 8, 1]     12
18  nagalakshmi         220  [8, 2, 7]     17
19       jyothi         221  [6, 2, 9]     17
```



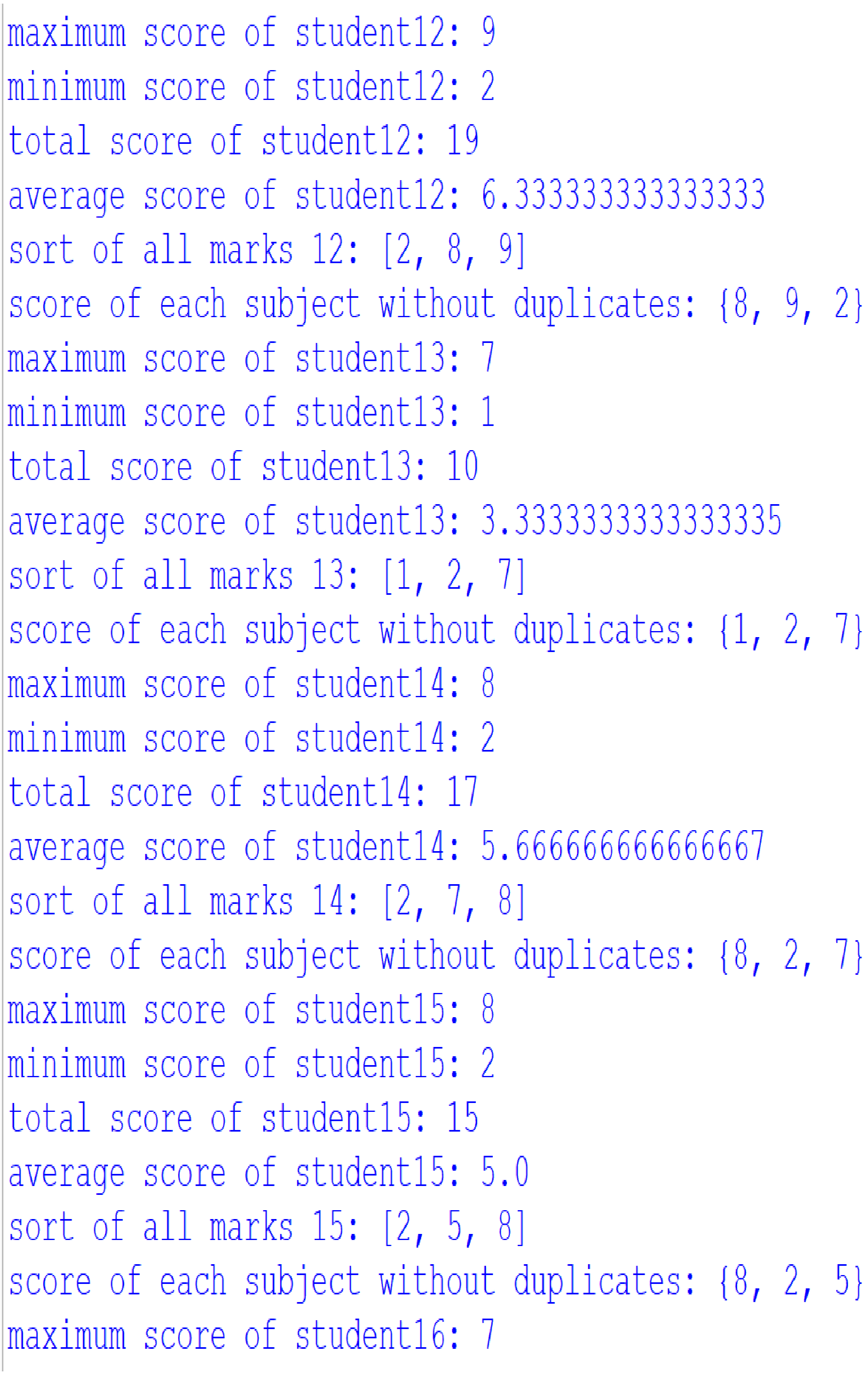
```
max total=: 19
min total=: 10
sort total=: [10, 10, 11, 12, 13, 15, 16, 16, 17, 17, 17, 17, 17, 17, 17, 18, 18, 18, 19, 19]
{10, 11, 12, 13, 15, 16, 17, 18, 19}
maximum score of student1: 8
minimum score of student1: 2
total score of student1: 17
average score of student1: 5.666666666666667
sort of all marks 1: [2, 7, 8]
score of each subject without duplicates: {8, 2, 7}
maximum score of student2: 9
minimum score of student2: 2
total score of student2: 17
average score of student2: 5.666666666666667
sort of all marks 2: [2, 6, 9]
score of each subject without duplicates: {9, 2, 6}
maximum score of student3: 8
minimum score of student3: 2
total score of student3: 16
average score of student3: 5.333333333333333
sort of all marks 3: [2, 6, 8]
score of each subject without duplicates: {8, 2, 6}
maximum score of student4: 8
```



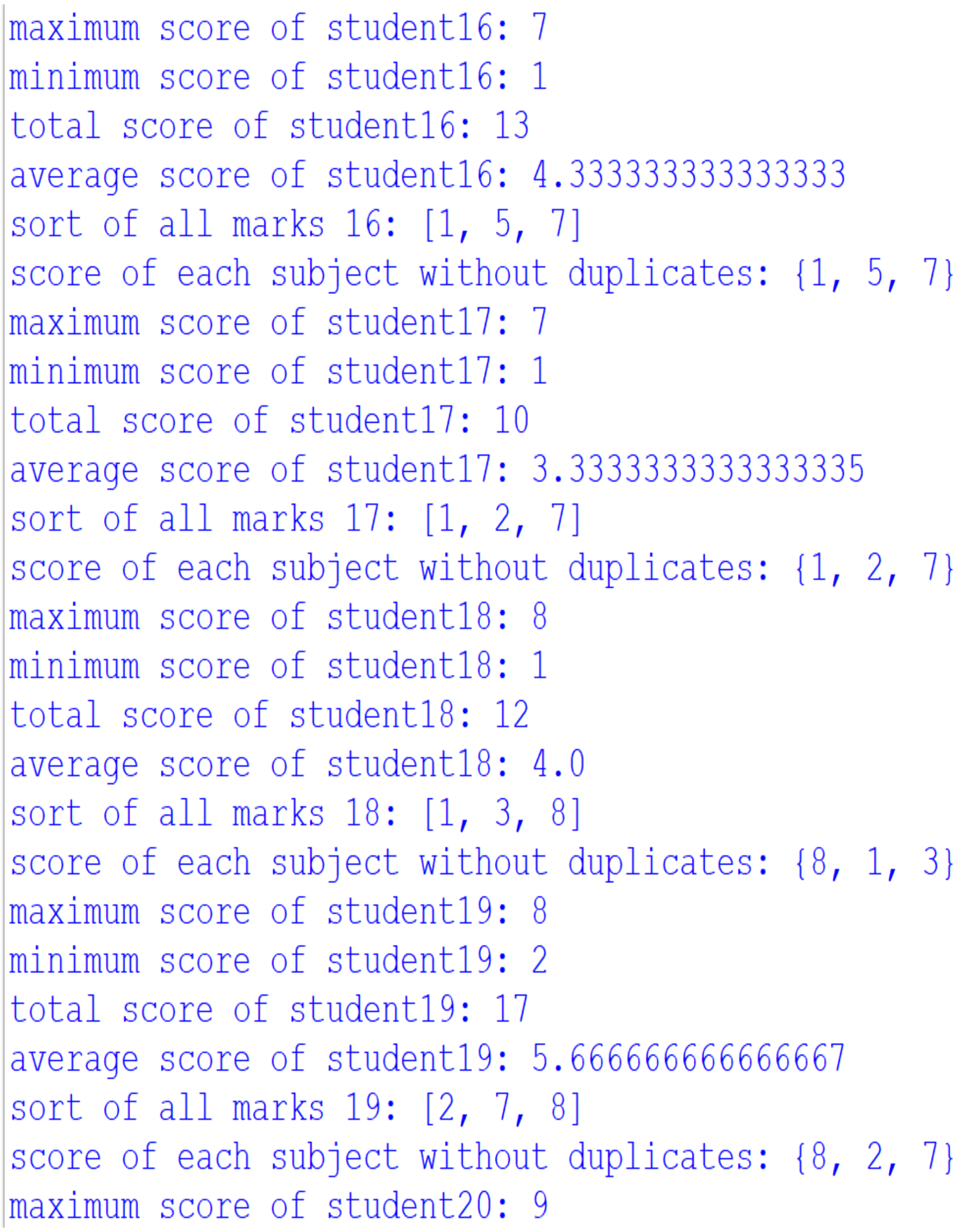
```
maximum score of student4: 8
minimum score of student4: 2
total score of student4: 17
average score of student4: 5.666666666666667
sort of all marks 4: [2, 7, 8]
score of each subject without duplicates: {8, 2, 7}
maximum score of student5: 9
minimum score of student5: 2
total score of student5: 18
average score of student5: 6.0
sort of all marks 5: [2, 7, 9]
score of each subject without duplicates: {9, 2, 7}
maximum score of student6: 9
minimum score of student6: 1
total score of student6: 17
average score of student6: 5.666666666666667
sort of all marks 6: [1, 7, 9]
score of each subject without duplicates: {1, 9, 7}
maximum score of student7: 8
minimum score of student7: 2
total score of student7: 16
average score of student7: 5.333333333333333
sort of all marks 7: [2, 6, 8]
score of each subject without duplicates: {8, 2, 6}
maximum score of student8: 9
```



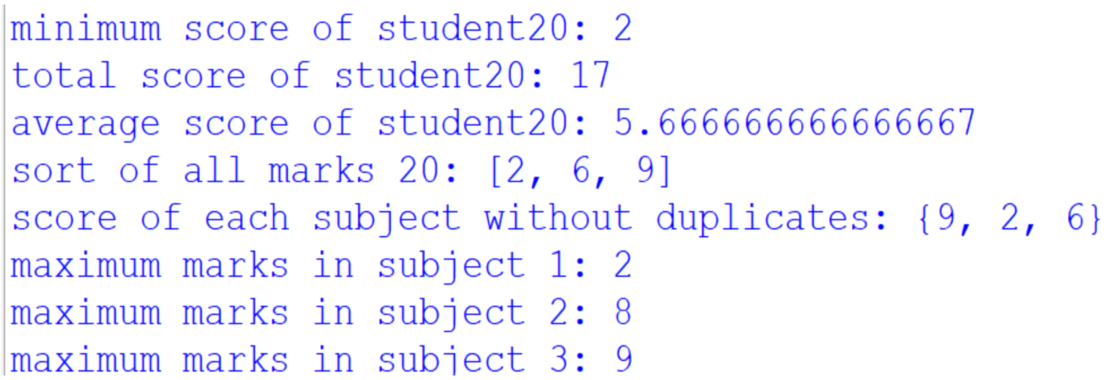
```
maximum score of student8: 9
minimum score of student8: 2
total score of student8: 18
average score of student8: 6.0
sort of all marks 8: [2, 7, 9]
score of each subject without duplicates: {9, 2, 7}
maximum score of student9: 9
minimum score of student9: 2
total score of student9: 18
average score of student9: 6.0
sort of all marks 9: [2, 7, 9]
score of each subject without duplicates: {9, 2, 7}
maximum score of student10: 8
minimum score of student10: 1
total score of student10: 11
average score of student10: 3.6666666666666665
sort of all marks 10: [1, 2, 8]
score of each subject without duplicates: {8, 1, 2}
maximum score of student11: 9
minimum score of student11: 2
total score of student11: 19
average score of student11: 6.333333333333333
sort of all marks 11: [2, 8, 9]
score of each subject without duplicates: {8, 9, 2}
maximum score of student12: 9
```



```
maximum score of student12: 9
minimum score of student12: 2
total score of student12: 19
average score of student12: 6.333333333333333
sort of all marks 12: [2, 8, 9]
score of each subject without duplicates: {8, 9, 2}
maximum score of student13: 7
minimum score of student13: 1
total score of student13: 10
average score of student13: 3.3333333333333335
sort of all marks 13: [1, 2, 7]
score of each subject without duplicates: {1, 2, 7}
maximum score of student14: 8
minimum score of student14: 2
total score of student14: 17
average score of student14: 5.666666666666667
sort of all marks 14: [2, 7, 8]
score of each subject without duplicates: {8, 2, 7}
maximum score of student15: 8
minimum score of student15: 2
total score of student15: 15
average score of student15: 5.0
sort of all marks 15: [2, 5, 8]
score of each subject without duplicates: {8, 2, 5}
maximum score of student16: 7
```



```
maximum score of student16: 7
minimum score of student16: 1
total score of student16: 13
average score of student16: 4.333333333333333
sort of all marks 16: [1, 5, 7]
score of each subject without duplicates: {1, 5, 7}
maximum score of student17: 7
minimum score of student17: 1
total score of student17: 10
average score of student17: 3.3333333333333335
sort of all marks 17: [1, 2, 7]
score of each subject without duplicates: {1, 2, 7}
maximum score of student18: 8
minimum score of student18: 1
total score of student18: 12
average score of student18: 4.0
sort of all marks 18: [1, 3, 8]
score of each subject without duplicates: {8, 1, 3}
maximum score of student19: 8
minimum score of student19: 2
total score of student19: 17
average score of student19: 5.666666666666667
sort of all marks 19: [2, 7, 8]
score of each subject without duplicates: {8, 2, 7}
maximum score of student20: 9
```



```
minimum score of student20: 2
total score of student20: 17
average score of student20: 5.666666666666667
sort of all marks 20: [2, 6, 9]
score of each subject without duplicates: {9, 2, 6}
maximum marks in subject 1: 2
maximum marks in subject 2: 8
maximum marks in subject 3: 9
```

Explaination:-

First we take N no of inputs, then we write a for loop to iterate our data N times, then we create a function in that take an empty set whenever we call the function name it redirected to the function and store the data, our stored data appended into an empty list s.we can extract our data using the index(s[i]) create a empty list then,write a for loop to store the student marks in each subject in that empty list.then by using min,max,len functions we have to find our required data.create another for loop to store marks of each subjects and store them in an empty list.then print our required output.

Conclusion:- our Python code for the student scholastic dashboard efficiently computes total marks, minimum and maximum scores per student, sorts subject marks, and ensures accurate summation without duplication. This comprehensive tool offers educators, students, and parents valuable insights into academic performance, facilitating targeted interventions and informed decision-making. Overall, our program streamlines academic assessment, aiding in optimizing student learning outcomes and fostering academic success.