Name: Adwait S Purao

UID: 2021300101

Batch: B2

Q1:

Define a class Cricketer which has:-

Attributes:-

- **player_name**
- **runs_hit**
- **innings_count**
- **not_out_count**
- **batting_avg**

Methods:-

- **get_avg**

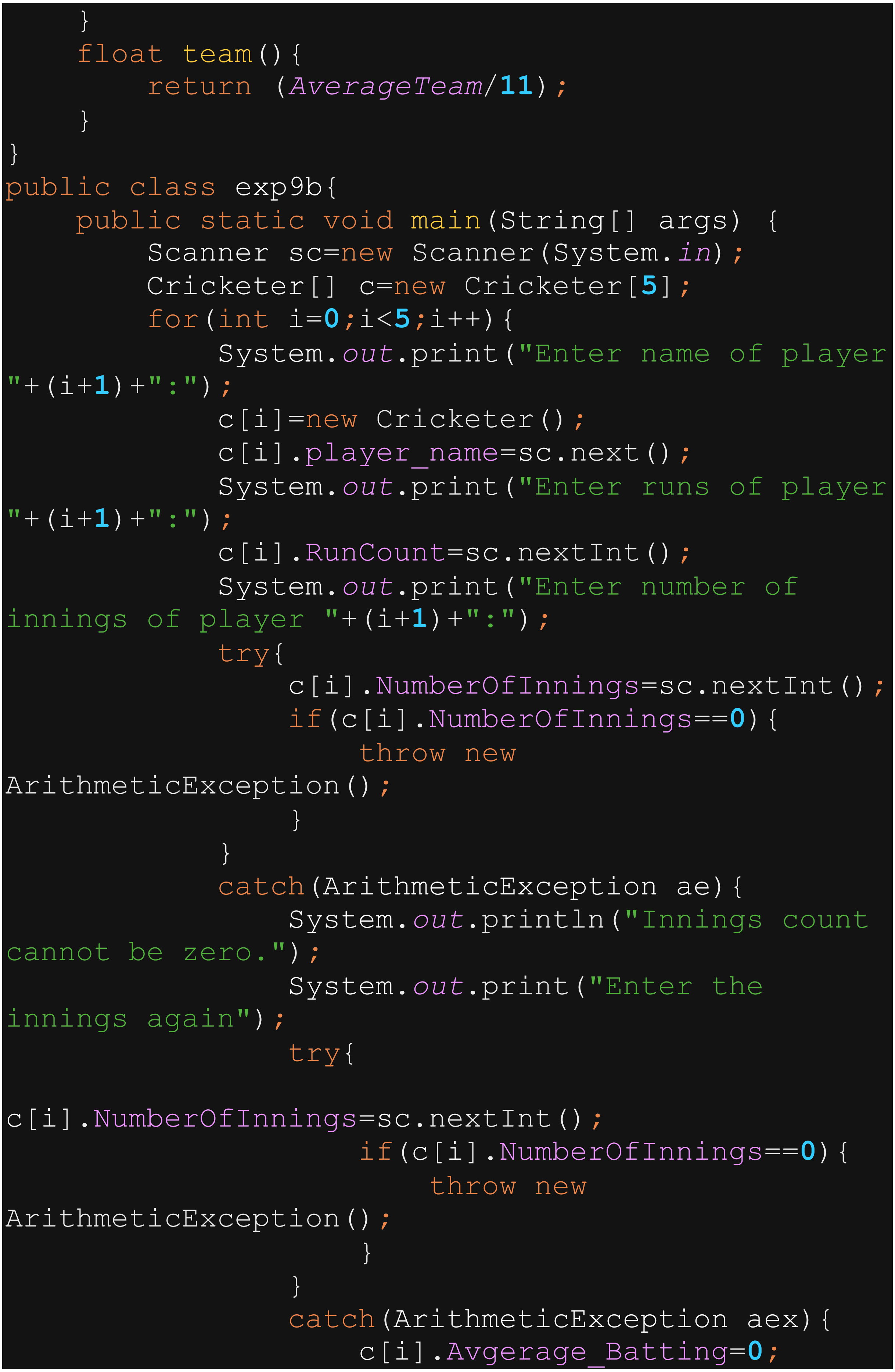
Make a cricket team with 11 cricketers. For each cricketer, find his batting average. Handle all different errors while calculating this. Also make a method which will find the list of cricketers in ascending order of their batting average and also display the cricketer stats in this order.

If the average of the batting average of the entire team is less than 20 runs then throw a user defined exception.

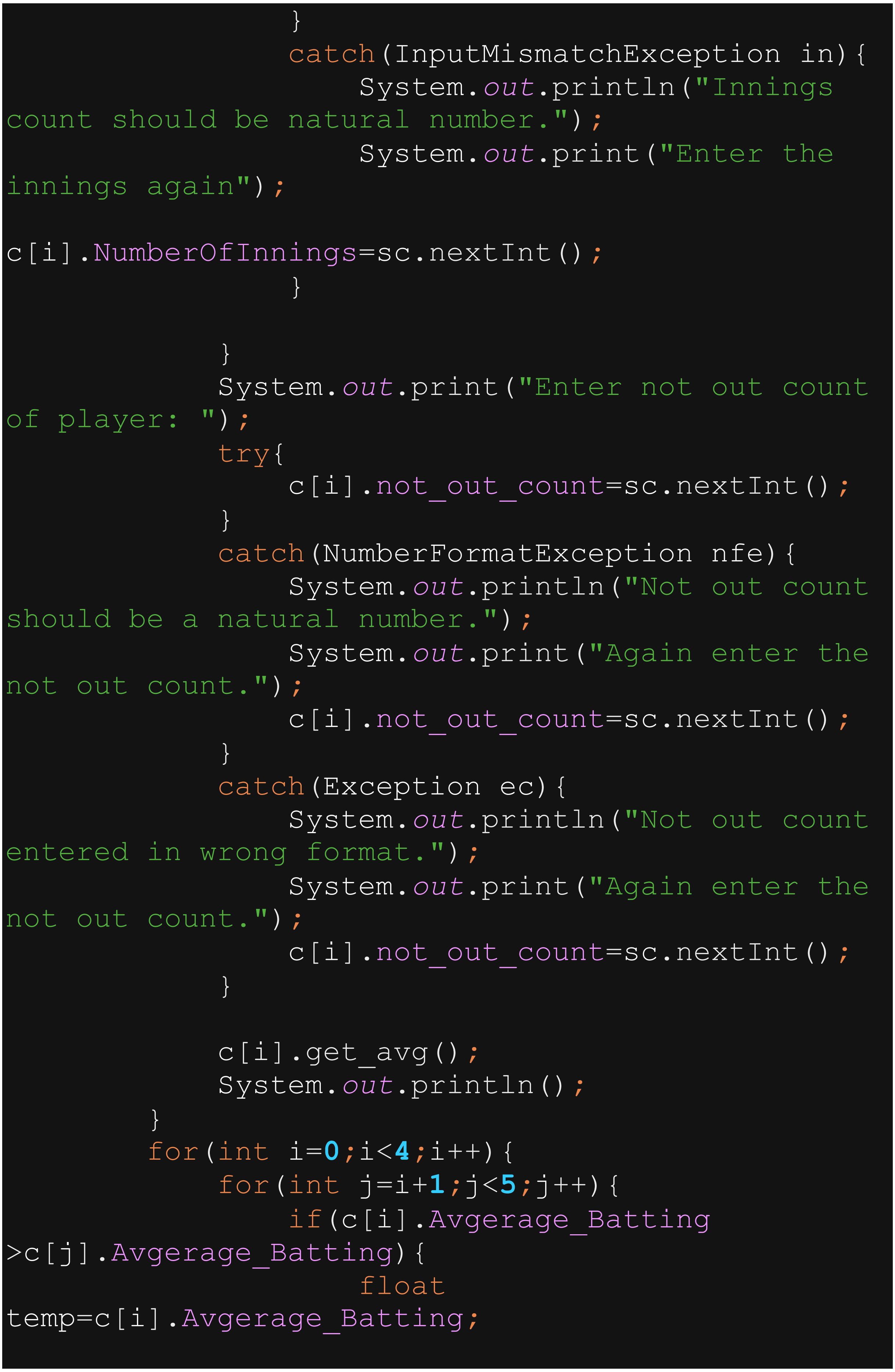
Note- Use the concept of Finallly block, mulitplt catch blocks and handle errors like ArrayIndexOutOfBoundsException, ArithmeticException, ArrayStoreException, NumberFormatException, etc

Code:

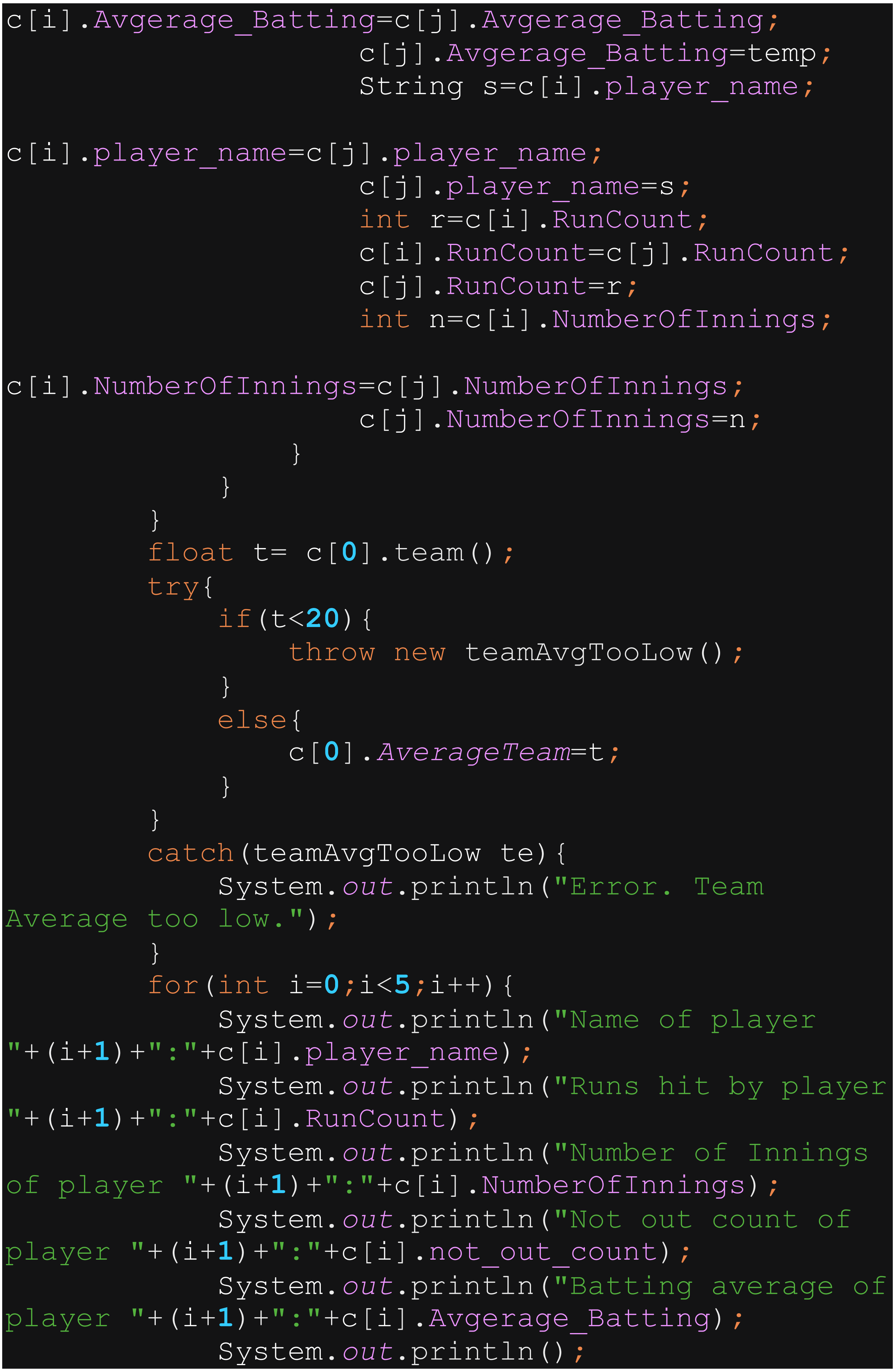
```
import java.util.*;
class teamAvgTooLow extends Exception{}
class Cricketer{
    String player_name;
    int RunCount, NumberOfInnings, not_out_count;
    float Avgerage_Batting;
    static float AverageTeam;
    void get_avg(){

this.Avgerage_Batting=(float)this.RunCount/(float)
this.NumberOfInnings;
        AverageTeam+=this.Avgerage_Batting;
```



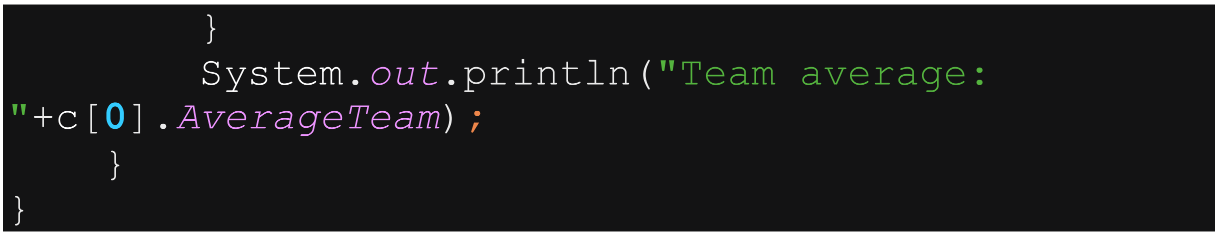
```
    }
    float team(){
        return (AverageTeam/11);
    }
}
public class exp9b{
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        Cricketer[] c=new Cricketer[5];
        for(int i=0;i<5;i++){
            System.out.print("Enter name of player
"+(i+1)+":");
            c[i]=new Cricketer();
            c[i].player_name=sc.next();
            System.out.print("Enter runs of player
"+(i+1)+":");
            c[i].RunCount=sc.nextInt();
            System.out.print("Enter number of
innings of player "+(i+1)+":");
            try{
                c[i].NumberOfInnings=sc.nextInt();
                if(c[i].NumberOfInnings==0){
                    throw new
ArithmeticException();
                }
            }
            catch(ArithmeticException ae){
                System.out.println("Innings count
cannot be zero.");
                System.out.print("Enter the
innings again");
                try{

c[i].NumberOfInnings=sc.nextInt();
                    if(c[i].NumberOfInnings==0){
                        throw new
ArithmeticException();
                    }
                }
                catch(ArithmeticException aex){
                    c[i].Avgerage_Batting=0;
```



```
                }
                catch(InputMismatchException in){
                    System.out.println("Innings
count should be natural number.");
                    System.out.print("Enter the
innings again");

c[i].NumberOfInnings=sc.nextInt();
                }

            }
            System.out.print("Enter not out count
of player: ");
            try{
                c[i].not_out_count=sc.nextInt();
            }
            catch(NumberFormatException nfe){
                System.out.println("Not out count
should be a natural number.");
                System.out.print("Again enter the
not out count.");
                c[i].not_out_count=sc.nextInt();
            }
            catch(Exception ec){
                System.out.println("Not out count
entered in wrong format.");
                System.out.print("Again enter the
not out count.");
                c[i].not_out_count=sc.nextInt();
            }

            c[i].get_avg();
            System.out.println();
        }
        for(int i=0;i<4;i++){
            for(int j=i+1;j<5;j++){
                if(c[i].Avgerage_Batting
>c[j].Avgerage_Batting){
                    float
temp=c[i].Avgerage_Batting;
```

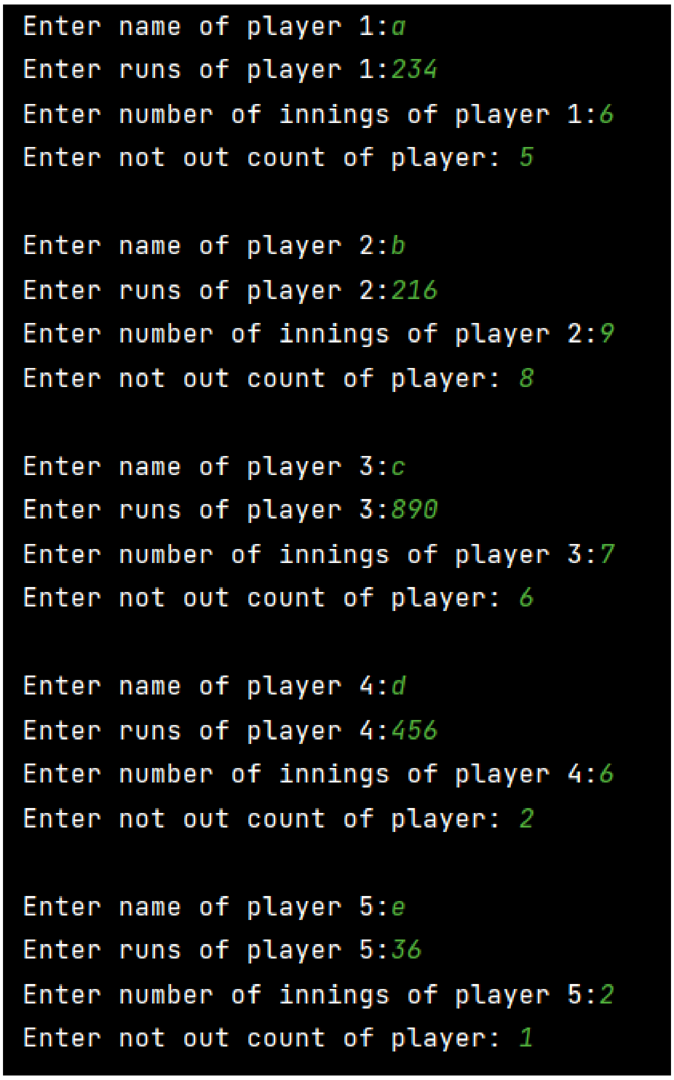


```
c[i].Avgerage_Batting=c[j].Avgerage_Batting;
                    c[j].Avgerage_Batting=temp;
                    String s=c[i].player_name;

c[i].player_name=c[j].player_name;
                    c[j].player_name=s;
                    int r=c[i].RunCount;
                    c[i].RunCount=c[j].RunCount;
                    c[j].RunCount=r;
                    int n=c[i].NumberOfInnings;

c[i].NumberOfInnings=c[j].NumberOfInnings;
                    c[j].NumberOfInnings=n;
                }
            }
        }
        float t= c[0].team();
        try{
            if(t<20){
                throw new teamAvgTooLow();
            }
            else{
                c[0].AverageTeam=t;
            }
        }
        catch(teamAvgTooLow te){
            System.out.println("Error. Team
Average too low.");
        }
        for(int i=0;i<5;i++){
            System.out.println("Name of player
"+(i+1)+":"+c[i].player_name);
            System.out.println("Runs hit by player
"+(i+1)+":"+c[i].RunCount);
            System.out.println("Number of Innings
of player "+(i+1)+":"+c[i].NumberOfInnings);
            System.out.println("Not out count of
player "+(i+1)+":"+c[i].not_out_count);
            System.out.println("Batting average of
player "+(i+1)+":"+c[i].Avgerage_Batting);
            System.out.println();
```

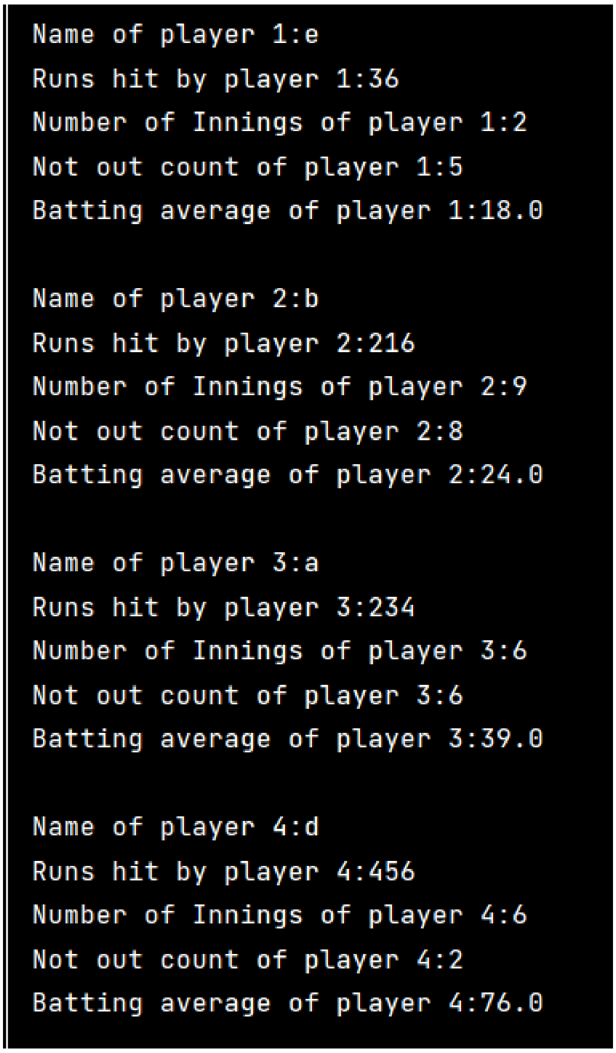


```
            }
            System.out.println("Team average: 
"+c[0].AverageTeam);
      }
}
```

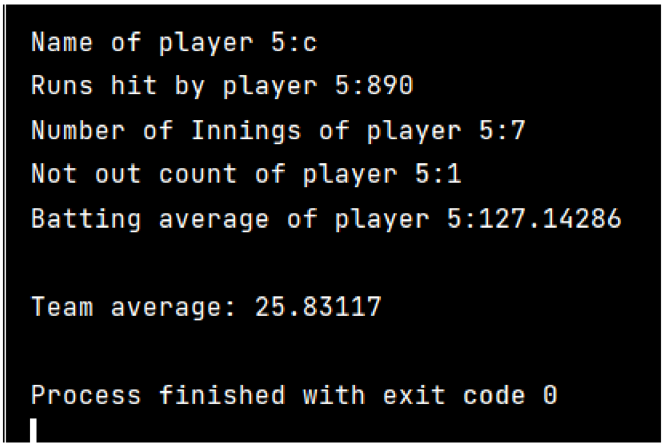
Output:



```
Enter name of player 1:a
Enter runs of player 1:234
Enter number of innings of player 1:6
Enter not out count of player: 5

Enter name of player 2:b
Enter runs of player 2:216
Enter number of innings of player 2:9
Enter not out count of player: 8

Enter name of player 3:c
Enter runs of player 3:890
Enter number of innings of player 3:7
Enter not out count of player: 6

Enter name of player 4:d
Enter runs of player 4:456
Enter number of innings of player 4:6
Enter not out count of player: 2

Enter name of player 5:e
Enter runs of player 5:36
Enter number of innings of player 5:2
Enter not out count of player: 1
```



```
Name of player 1:e
Runs hit by player 1:36
Number of Innings of player 1:2
Not out count of player 1:5
Batting average of player 1:18.0

Name of player 2:b
Runs hit by player 2:216
Number of Innings of player 2:9
Not out count of player 2:8
Batting average of player 2:24.0

Name of player 3:a
Runs hit by player 3:234
Number of Innings of player 3:6
Not out count of player 3:6
Batting average of player 3:39.0

Name of player 4:d
Runs hit by player 4:456
Number of Innings of player 4:6
Not out count of player 4:2
Batting average of player 4:76.0
```



```
Name of player 5:c
Runs hit by player 5:890
Number of Innings of player 5:7
Not out count of player 5:1
Batting average of player 5:127.14286

Team average: 25.83117

Process finished with exit code 0
```