# Practical no:1

'''Group A1 in second year computer engg class, group A students play cricket, group B students play badminton and group C students play football. Write a python program using functions to compute the following:

1)List of students who play both cricket and badminton.

2)List of students who play either cricket or badminton but not both.

3)Number of student who play neither cricket nor badminton.

4)Number of student who play cricket and football but not badminton.

NOTE:While realizing the group, duplicate entry should be avoided, do not use set built in function.'''

def main(): cricket=[] badminton=[] football=[] while True: print("1)ACCEPT DATA.") print("2)DISPLAY DATA.")

print("3)LIST OF STUDENT WHO PLAY BOTH CRICKET AND BADMINTON.") print("4)LIST OF STUDENT WHO PLAY EITHER CRICKET OR BADMINTON BUT NOT BOTH.")

print("5)NUMBER OF STUDENT WHO PLAY NEITHER CRICKET NOR BADMINTON.")

print("6)NUMBER OF STUDENT WHO PLAY CRICKET AND FOOTBALL BUT NOT BADMINTON.")

print("7)END.")

print("ENTER YOUR CHOICE:")

choice=int(input()) if(choice==1):

print("ENTER DATA FOR CRICKET:") Accept(cricket)

print("ENTER DATA FOR BADMINTON:") Accept(badminton)

print("ENTER DATA FOR FOOTBALL:")

Accept(football) elif(choice==2): print("DATA FOR CRICKET:") print(cricket)

print("DATA FOR BADMINTON:") print(badminton) print("DATA FOR FOOTBALL:") print(football) elif(choice==3):

cricket\_badminton(cricket,badminton) elif(choice==4):

either\_cricket\_badminton(cricket,badminton)

elif(choice==5):

neither\_cricket\_nor\_badminton(cricket,badminton,football) elif(choice==6):

either\_cricket\_football\_nor\_badminton(cricket,badminton,football) elif(choice==7):

break;

def Accept(player):

n=int(input("ENTER HOW MANY PLAYERS: ")) for i in range(n):

x=input("ENTER NAME: ")

player.append(x)

def cricket\_badminton(cricket,badminton):

cr\_bad=[] for i in range(len(cricket)):

x=cricket[i] if x in badminton: cr\_bad.append(x)

print("LIST OF STUDENTS WHO PLAY BOTH CRICKET AND BADMINTON: ") print(cr\_bad)

def either\_cricket\_badminton(cricket,badminton): final=[] for a in range(len(cricket)): final.append(cricket[a]) output=[] for i in range(len(badminton)): if(badminton[i] not in cricket): final.append(badminton[i]) Intersection=[] for i in range(len(cricket)): if(cricket[i] in badminton): Intersection.append(cricket[i]) for i in range(len(final)): if(final[i] not in Intersection):

output.append(final[i])

print("LIST OF STUDENTS WHO PLAY EITHER CRICKET OR BADMINTON BUT NOT BOTH: ")

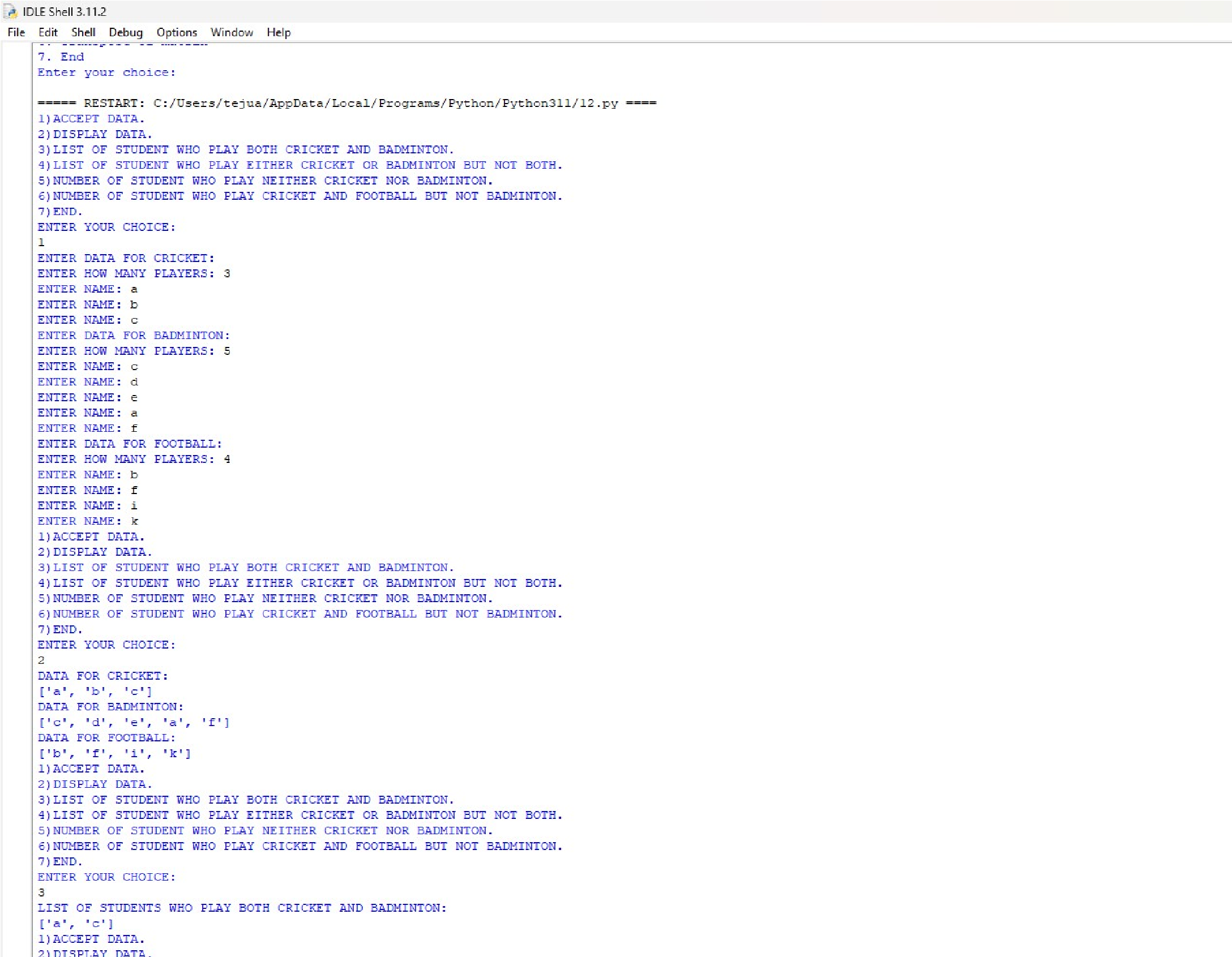
print(output)

def neither\_cricket\_nor\_badminton(cricket,badminton,football): final=[] for i in range(len(cricket)): final.append(cricket[i]) for i in range(len(badminton)): if(badminton[i] not in final): final.append(badminton[i]) output=[]

for i in range(len(football)): if(football[i] not in final):

output.append(football[i])

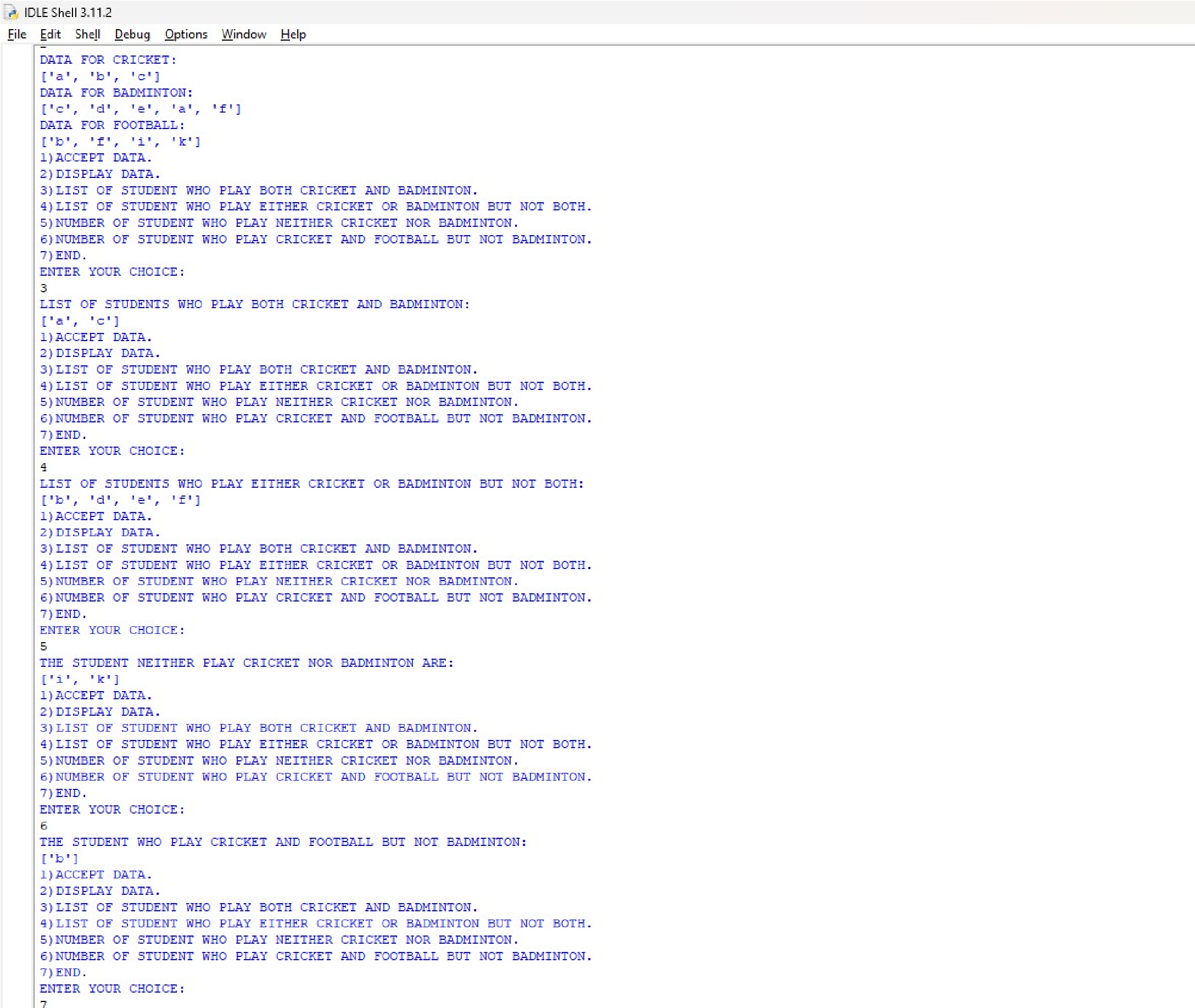
print("THE STUDENT NEITHER PLAY CRICKET NOR BADMINTON ARE: ") print(output)

def either\_cricket\_football\_nor\_badminton(cricket,badminton,football): final=[] output=[] for i in range(len(cricket)): if (cricket[i] in football): final.append(cricket[i]) for i in range(len(final)): output.append(final[i])

print("THE STUDENT WHO PLAY CRICKET AND FOOTBALL BUT NOT BADMINTON: ") print(output)

main()

o/P:



# Practical no:2

'''Write a python program to store marks scored in subject fundamental of data structure by N students in class.Write function to compute following:

1)The average score of class.

2)Highest score and lowest score of class.

3)Count of student who were absent for the test. 4)Display marks with highest frequency''' def main(): fds=[] while True:

print("1)Enter Marks") print("2)Display Marks") print("3)Average of Marks") print("4)Highest score and lowest Score") print("5)Absent student Count") print("6)Display marks with Highest Frequency") print("7)End") print("Enter your choice: ") choice=int(input()) if (choice==7): break elif(choice==1):

accept\_marks(fds) elif(choice==2):

print("FDS Marks are: ")

print(fds) elif(choice==3):

average\_marks(fds) elif(choice==4):

highest\_lowest\_marks(fds) elif(choice==5):

absent\_count(fds) elif(choice==6):

display\_marks\_with\_highest\_frequency(fds) def accept\_marks(fds): print("Enter number of students: ")

n=int(input()) for i in range(n): print("Enter marks: ") x=int(input()) fds.append(x) def average\_marks(fds):

total=0 present=0 for i in fds: if(i!=-1):

total=total+i

present=present+1 average\_marks=total/present print("Average is: ",average\_marks) def highest\_lowest\_marks(fds):

highest=-1 lowest=999 for i in fds: if(i!=-1): if(i>highest): highest=i if(i<lowest): lowest=i print("Highest Marks is: ",highest) print("Lowest Marks is: ",lowest) def absent\_count(fds): count=0 for i in fds: if(i==-1):

count=count+1 print("Absent student: ") print(count) def display\_marks\_with\_highest\_frequency(fds):

freq=0 for i in range(len(fds)):

count=0 if(i!=-1): for j in range(len(fds)): if (fds[i]==fds[j]): count=count+1 if(freq<count): freq=count high\_freq\_mark=fds[i] print("Highest frequency marks is ",high\_freq\_mark) print("Highest frequency count is ",freq) main()

o/P:





# Practical no:4

A)'''Write a python program to store roll no of students in array who attended training program in random order. Write function for searching whether particular student attending training program or not using linear search and sentinel search''' def main(): Roll\_no=[] while True:

print("1. Accept Roll no") print("2. Display Roll no") print("3. Linear Search") print("4. Sentinal Search") print("5. End") print("Enter the choice: ") choice=int(input()) if(choice==1):

Accept\_Rollno(Roll\_no) elif(choice==2):

print("Entered Roll no are: ",Roll\_no) elif(choice==3):

Linear\_search(Roll\_no) elif(choice==4):

Sentinal\_search(Roll\_no) elif(choice==5):

break def Accept\_Rollno(Roll\_no):

n=int(input("Enter the total number of students: ")) for i in range(n):

x=int(input("Enter roll no: "))

Roll\_no.append(x)

def Linear\_search(Roll\_no): n=len(Roll\_no) search\_key=int(input("Enter roll no to be searched: ")) flag=0 for i in range(n): if(search\_key==Roll\_no[i]):

print("Roll no is found in training student") flag=1 break if(flag==0):

print("Roll no is not found in training")

def Sentinal\_search(Roll\_no): n=len(Roll\_no) search\_key=int(input("Enter roll no to be searched: ")) last=Roll\_no[n-1] Roll\_no[n-1]=search\_key

i=0 while(Roll\_no[i]!=search\_key):

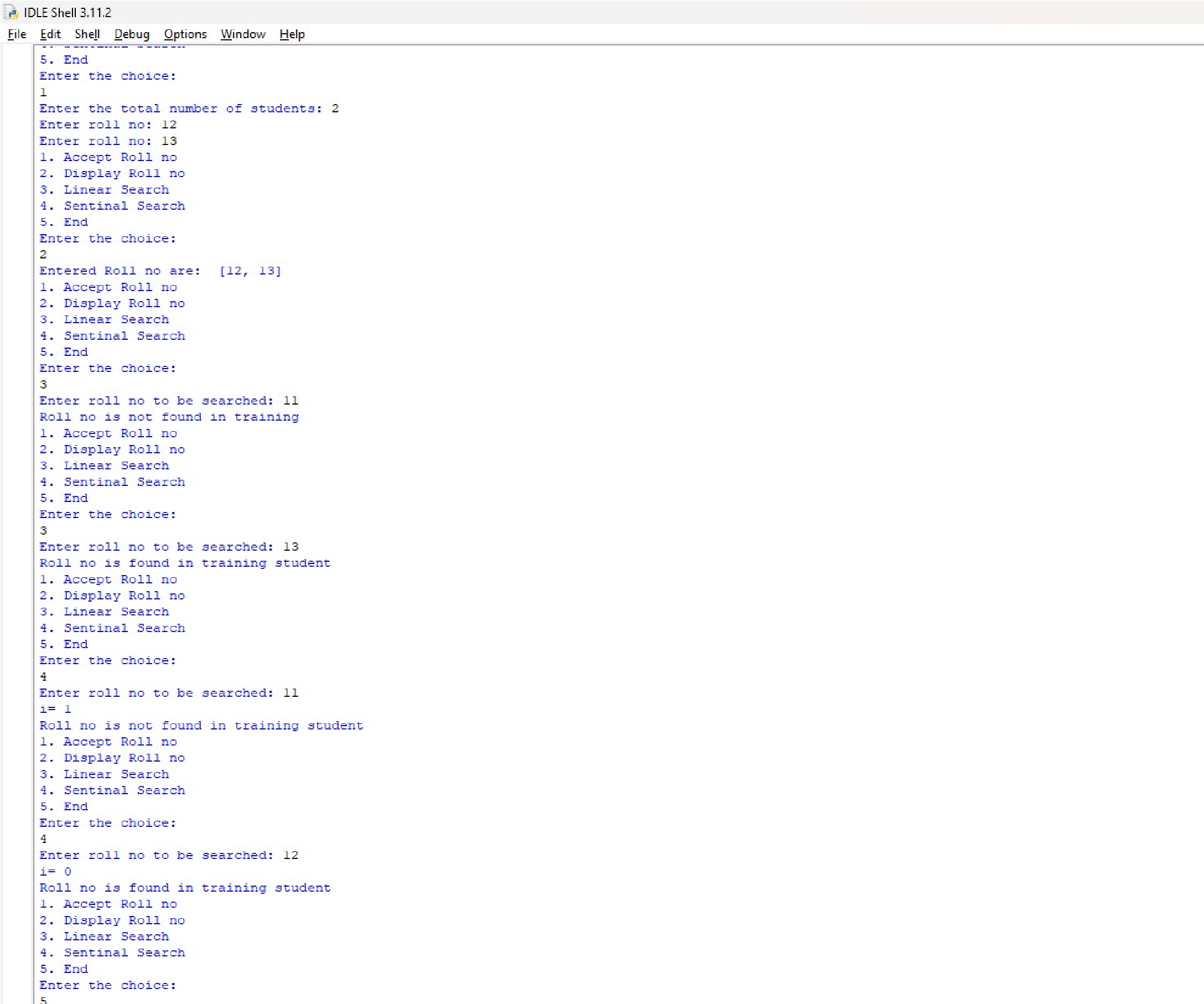
i=i+1 print("i=",i)

Roll\_no[n-1]=last if((search\_key==Roll\_no[n-1])or(i<n-1)): print("Roll no is found in training student") else:

print("Roll no is not found in training student") main()

o/P:





B)'''Write a python program to store roll no of students in array who attended training program in sorted order. Write function for searching whether particular student attending training program or not using binary search''' def main(): Roll\_no=[] while True:

print("1. Accept roll no in sorted order") print("2. Display roll no in sorted order") print("3. Iterative Binary Search") print("4. Recursive Binary Search")

print("5. End") print("Enter your choice: ") choice=int(input()) if(choice==5): break elif(choice==1):

Accept\_Rollno(Roll\_no) elif(choice==2):

print("Entered sorted roll no are: ",Roll\_no) elif(choice==3):

Iterative\_binary\_search(Roll\_no) elif(choice==4):

low=0

high=len(Roll\_no)-1 key=int(input("Enter Roll no to be searched: ")) Recursive\_binary\_search(Roll\_no,low,high,key)

def Accept\_Rollno(Roll\_no):

n=int(input("Enter the total no of students: ")) for i in range(n):

x=int(input("Enter roll no: "))

Roll\_no.append(x)

def Iterative\_binary\_search(Roll\_no): low=0 high=len(Roll\_no)-1 key=int(input("Enter roll no to be searched: ")) while(low<=high):

mid=int((low+high)/2) if (Roll\_no[mid]==key):

print("Roll no found in training") return elif(Roll\_no[mid]<key):

low=mid+1 else:

low=mid-1 if(low>high):

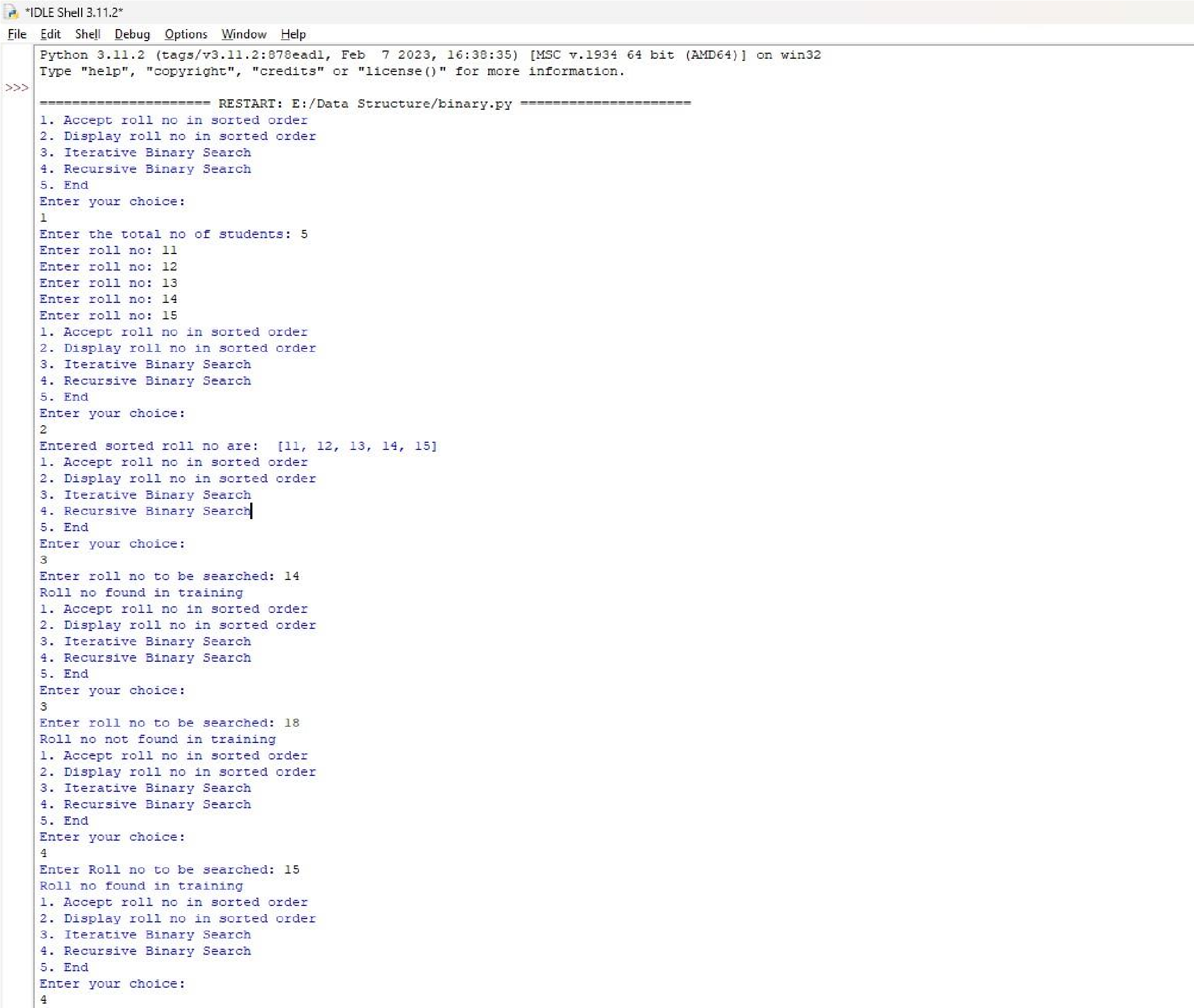
print("Roll no not found in training")

def Recursive\_binary\_search(Roll\_no,low,high,key): if(low<=high):

mid=int((low+high)/2) if (Roll\_no[mid]==key):

print("Roll no found in training")

return elif(Roll\_no[mid]<key):

 Recursive\_binary\_search(Roll\_no,mid+1,high,key) else:

Recursive\_binary\_search(Roll\_no,low,mid-1,key) if(low>high):

print("Roll no not found in training")

main()

o/P:

