

Examination Roll No. → 20234757062

Name → Anjali Singh

Date & Time of Examination → 27/03/21
9:30 - 12:30

Unique Paper Code → 223401101

Name of Programme → MCA (CBCS PG)

Semester → Ist

Title of the Paper → Object Oriented Programming

Email id → 200105@cs.du.ac.in

Mobile Number → 8770061023

Total Pages used in Question → 2

Question Num → 4

def iteration(k, lst):

 "Input: list and length/size of list to be sorted at that size"

 output: update list by reference after sorting"

 if k == len(lst): # if k is size of given list
 raise Exception("No more iterations can be done")

 # raise Exception because we can't do further operation.

else:

 minEle = float('inf')

 for i in lst[k:]: # min element finding
 minEle = min(minEle, i)

 minPosition = lst.index(minEle)

 # showing minEle index

 t = lst[k] # exchanging element by

 lst[k] = lst[minPosition] # help of t variable

 lst[minPosition] = t

def main(): # main function

 n = int(input("Enter size of list"))

 lst = list(map(int, input().split(" ")))

 # taking user input for list

k = 0

while (True): # for menu driven

print("Menu /n 1. To execute next iteration
/n 2. To show status /n 3. Exit")

choice = int(input("Enter your choice"))

if (choice == 1):

try: # exception handling

iteration(k, lst)

except Exception as E:

print(E)

break

k += 1

elif (choice == 2):

print(f"Iteration to be execute : {k}, {lst}")
print(f"Result after iteration {lst}")
print using f string

elif (choice == 3):

break

if __name__ == "__main__":

main() # calling main function

Assumptions : As Question is Ambiguous
doing it by swapping rather
than shifting