Lab 6 — Functions & List Manipulations

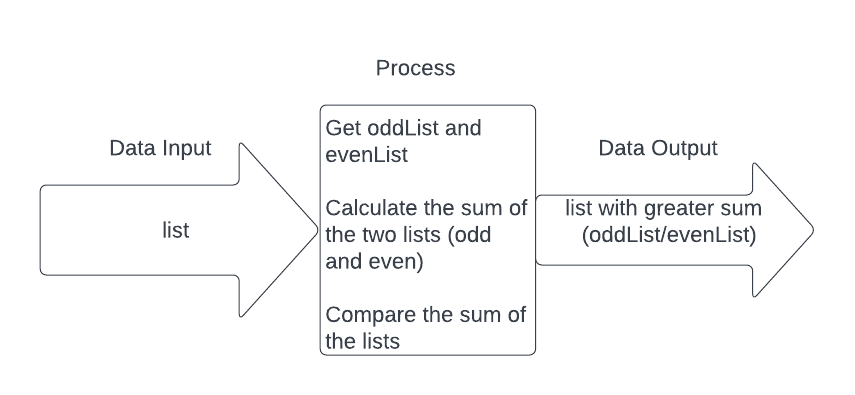
Name

Institution

1. Program

| def check\_odd\_even(lst):  """  Takes a list and creates two lists one with elements whose  indices are even and the other list whose indices are odd.  :param lst:  :return: odd\_list, even\_list  """   *# initialize empty odd and even list*  odd\_list = []  even\_list = []  for i in range(len(lst)): *# loop through the indices of the elements in the list*  if i % 2 == 0: *# check if index is even by using modulus*  even\_list.append(lst[i]) *# append elements with even index to the even list*  else:  odd\_list.append(lst[i]) *# append elements with odd index to the odd list*  return odd\_list, even\_list *# return both odd and even lists*   def greaterSumList(oddList, evenList):  """   :param oddList:  :param evenList:  :return: either oddList or evenList  """  sumOdd = sum(oddList) *# get the sum of elements in the odd list*  sumEven = sum(evenList) *# get the sum of elements in the even list*  if sumEven > sumOdd: *# check if sum of even list is greater than sum of odd list*  return evenList *# return even list if condition is true*  else:  return oddList *# return odd list if condition is false*   def main():  """  Driver function  """  lst = [1, 4, 9, 16, 25, 28, 36, 33, 49, 64, 100, 81, 33] *# use the list in the guide*  odd, even = check\_odd\_even(lst) *# get both odd and even lists*  returned\_lst = greaterSumList(odd, even) *# get the list with a greter sum*  print(returned\_lst) *# print the list with a greater sum*   if \_\_name\_\_ == "\_\_main\_\_":  main() |
| --- |

1. Program design



1. Flowchart

