CS Term – 2 Practical's

CS Practical file of Sushant for Term – 2

Sushant Sejwal

Class - XI -A
Roll no. – 46

Teacher – Pawanjeet Singh

Practical 1

Write a Python Program to find the largest/smallest number in a list/tuple and displaying the position of that number

Code

```
lst = []
is_sushant_awesome = True

while is_sushant_awesome:
    lst_ele = input('Enter element for list or type "quit()" to leave\n ->  ')

    if lst_ele.lower().strip() == 'quit()':
        break

    elif lst_ele.isdigit():
        lst_ele = int(lst_ele)
        lst.append(lst_ele)
    else:
        print('Enter only numbers')
    print()

print(f'list -> {lst}')

# Finding the largest number in the list
lar_num = 0
for i in lst:
    if i > lar_num:
        lar_num = i
lar_num_pos = lst.index(lar_num)
print()
print(f'the largest number in the lst is {lar_num} and it\'s index position is
{lar_num_pos}')


# Finding the smallest number in the list
sml_num = lar_num
for i in lst:
    if i < sml_num:
        sml_num = i
sml_num_pos = lst.index(sml_num)
print(f'the smallest number in the lst is {sml_num} and it\'s index position is
{sml_num_pos}')
```

```
# Output
# Enter element for list or type "quit()" to leave
#  ->  23

# Enter element for list or type "quit()" to leave
#  ->  65

# Enter element for list or type "quit()" to leave
#  ->  quit()
# list -> [23, 65]

# the largest number in the lst is 65 and it's index position is 1
# the smallest number in the lst is 23 and it's index position is 0
```

Practical 2

Write a Python Program to Input a list of numbers and swap elements at the even location with the elements at the odd location.

Code

```
lst = []
is_sushant_awesome = True

while is_sushant_awesome:
    lst_ele = input('Enter element for list or type "quit()" to leave\n ->  ')

    if lst_ele.lower().strip() == 'quit()':
        break

    elif lst_ele.isdigit():
        lst_ele = int(lst_ele)
        lst.append(lst_ele)
    else:
        lst.append(lst_ele)
    print()

print(f'original list -> {lst}')
print()

if len(lst) % 2 ==0:
    range = range(0, len(lst), 2)
else:
```

```
    range = range(0, len(lst)-1, 2)

for i in range:
    lst[i], lst[i+1] = lst[i+1], lst[i]

print(f'list after swaping elements positions -> {lst}')

# Output
# Enter element for list or type "quit()" to leave
#  ->  324

# Enter element for list or type "quit()" to leave
#  ->  34

# Enter element for list or type "quit()" to leave
#  ->  2

# Enter element for list or type "quit()" to leave
#  ->  768

# Enter element for list or type "quit()" to leave
#  ->  098

# Enter element for list or type "quit()" to leave
#  ->  23

# Enter element for list or type "quit()" to leave
#  ->  quit()
# original list -> [324, 34, 2, 768, 98, 23]

# list after swaping elements positions -> [34, 324, 768, 2, 23, 98]
```

Practical 3

Write a Python Program to Input a list/tuple of elements, search for a given element in the list/tuple and frequency and position of search element.

Code

```
lst = []
is_sushant_awesome = True

while is_sushant_awesome:
    lst_ele = input('Enter element for list or type "quit()" to leave\n ->  ')
```

```
    if lst_ele.lower().strip() == 'quit()':
        break

    else:
        lst.append(lst_ele)
    print()

srch = input('Enter the element you want to search in the list\n ->  ')

if srch in lst:
    srch_count = lst.count(srch)
    srch_pos = lst.index(srch)
    print(f"{srch} has appeared in the list {srch_count} times and at {srch_pos}
index position")

else:
    print(f"{srch} doesn't exist in the list {lst}")

# Output
# Enter element for list or type "quit()" to leave
#  ->  234

# Enter element for list or type "quit()" to leave
#  ->  345

# Enter element for list or type "quit()" to leave
#  ->  23

# Enter element for list or type "quit()" to leave
#  ->  56

# Enter element for list or type "quit()" to leave
#  ->  2424

# Enter element for list or type "quit()" to leave
#  ->  quit()
# Enter the element you want to search in the list
#  ->  23
# 23 has appeared in the list 1 times and at 2 index position
```

Practical 4

Write a Python Program to Input a list of numbers and test if a number is equal to the sum of the cubes of its digits. Find the smallest and largest such number from the given list of numbers.

Code

```
lst = []
is_sushant_awesome = True

while is_sushant_awesome:
    lst_ele = input('Enter element for list or type "quit()" to leave\n ->  ')

    if lst_ele.lower().strip() == 'quit()':
        break

    elif lst_ele.isdigit():
        lst_ele = int(lst_ele)
        lst.append(lst_ele)
    else:
        print(f"Enter only numbers")
    print()

for number in lst:
    sum_of_cubes = 0
    num = number
    while num > 0:
        num_last = num % 10
        sum_of_cubes = sum_of_cubes + num_last**3
        num = num // 10

    if sum_of_cubes == number:
        print(f'the sum of cubes of digits of number {number} is eaual to itself')
    else:
        print(f'the sum of cubes of digits of number {number} is not eaual to
itself')


# Finding the largest number in the list
lar_num = 0
for i in lst:
    if i > lar_num:
        lar_num = i
lar_num_pos = lst.index(lar_num)
print()
```

```
print(f'the largest number in the lst is {lar_num} and it\'s index position is
{lar_num_pos}')


# Finding the smallest number in the list
sml_num = lar_num
for i in lst:
    if i < sml_num:
        sml_num = i
sml_num_pos = lst.index(sml_num)
print(f'the smallest number in the lst is {sml_num} and it\'s index position is
{sml_num_pos}')

# Output
# Enter element for list or type "quit()" to leave
#  ->  324

# Enter element for list or type "quit()" to leave
#  ->  23

# Enter element for list or type "quit()" to leave
#  ->  324

# Enter element for list or type "quit()" to leave
#  ->  54

# Enter element for list or type "quit()" to leave
#  ->  22

# Enter element for list or type "quit()" to leave
#  ->  quit
# Enter only numbers

# Enter element for list or type "quit()" to leave
#  ->  quit()
# the sum of cubes of digits of number 324 is not eaual to itself
# the sum of cubes of digits of number 23 is not eaual to itself
# the sum of cubes of digits of number 324 is not eaual to itself
# the sum of cubes of digits of number 54 is not eaual to itself
# the sum of cubes of digits of number 22 is not eaual to itself

# the largest number in the lst is 324 and it's index position is 0
# the smallest number in the lst is 22 and it's index position is
```

Practical 5

Write a Python Program to Create a dictionary with the roll number, name and marks of n students in a class and display the names of students who have marks above 75.

Code

```
# Creating  awesome flag
is_sushant_awesome = "may be"
marks_above_75 = False

Class_sd = {}

while is_sushant_awesome:

    # this will run when there will be no element in dict class_sd
    if not Class_sd:
        yes_no = input('The Dictionary Class is empty. do wanna enter students data
in that, type yes[y] to enter or no[n] to leave\n =>  ')
        print() # printing a empty line for formatting output

    elif Class_sd:
        yes_no = input('Do wanna enter more students data in Dictionary Class, type
yes[y] to enter or no[n] to leave\n =>  ')
        print() # printing a empty line for formatting output


    yes_no = yes_no.lower().strip()

    if yes_no == 'no' or yes_no =='n':
        break

    elif yes_no == 'yes' or yes_no == 'y':
        std_name = input('Enter the name of student\n =>  ')
        std_roll = input(f'Enter the roll number of {std_name}\n =>  ')
        std_marks = int(input(f'Enter marks of {std_name} out of 100\n =>  '))
        if std_marks > 75:
            marks_above_75 = True
        if std_marks > 100 or std_marks < 0:
            std_marks = int(input('WHOOPSE Enter marks between 0 to 100\n =>  '))


        Class_sd[std_name] = dict([('Roll No',std_roll),('Marks',std_marks)])
        print() # Empty line for formatting
```

```
    else:
        print('WHOOPPSEE try again')
        continue


if marks_above_75:
    print('Students who get marks above 75 are :')
    for names in Class_sd:
        if Class_sd[names]['Marks'] > 75:
            print(f' Name - {names}, marks - {Class_sd[names]["Marks"]}')

else:
    print('There\'s no one in the class who got marks above 75')

# output
# The Dictionary Class is empty. do wanna enter students data in that, type yes[y] to
enter or no[n] to leave
#  =>  y

# Enter the name of student
#  =>  Sushant
# Enter the roll number of Sushant
#  =>  34
# Enter marks of Sushant out of 100
#  =>  99

# Do wanna enter more students data in Dictionary Class, type yes[y] to enter or no[n]
to leave
#  =>  y

# Enter the name of student
#  =>  Programiz
# Enter the roll number of Programiz
#  =>  35
# Enter marks of Programiz out of 100
#  =>  100

# Do wanna enter more students data in Dictionary Class, type yes[y] to enter or no[n]
to leave
#  =>  y

# Enter the name of student
#  =>  Jinex
# Enter the roll number of Jinex
#  =>  35
```

```
# Enter marks of Jinex out of 100
#  =>  100

# Do wanna enter more students data in Dictionary Class, type yes[y] to enter or no[n]
to leave
#  =>  n

# Students who get marks above 75 are :
#  Name - Sushant, marks - 99
#  Name - Programiz, marks - 100
#  Name - Jinex, marks - 100
```

Practical 6

Write a Python Program to Create a dictionary of employee information (empid, empname,empsal,empmobile), input five values from the user and display the employee who is getting salary less than or equal to Rs. 50000.

Code

```
emp_information = {}

emp_sal_above_50_k = False  # flag for employee salary

# Creating flag
is_sushant_awesome = 'any doubt'

while is_sushant_awesome:

    # If dictionary is empty then this will run
    if not emp_information:
        # asking user to add information or leave
        yes_or_no = input("Employee Information dictionary is empty. If you want to
add information then type yes[y] or type no[N] to leave\n =>  ")
        print() # empty line

    # If dictionary is not empty then this will run
    elif emp_information:
        yes_or_no = input("\nIf you want to add more entries to the employee
dictionary then type yes[y] or type no[N] to leave\n =>  ")
        print() # empty line

    else:
        print('Whoppse try again')
```

```
    # Breaking break
    if yes_or_no.lower().strip() == 'no' or yes_or_no.lower().strip() == 'n':
        break

    elif yes_or_no.lower().strip() == 'yes' or yes_or_no.lower().strip() == 'y':

        emp_name = input('Enter empoyee Name\n =>  ')
        emp_id = input(f'Enter {emp_name} ID\n =>  ')
        emp_sal = int(input(f'Enter {emp_name} Salary\n =>  '))

        if emp_sal <= 50_000:
            emp_sal_above_50_k = True

        emp_mob_no = int(input(f'Enter {emp_name} Mobile Number\n =>  '))
        # Checking mobile number
        if len(str(emp_mob_no)) > 10 or len(str(emp_mob_no)) < 10:
            print('Mobile no. shouold be of 10 digits only')
            emp_mob_no = int(input(f'Enter {emp_name} Mobile Number\n =>  '))

        # adding item to dict
        emp_information[emp_name] = dict([('ID',emp_id), ('Salary',emp_sal), ('Mobile
No.',emp_mob_no)])

    else:
        print('Whoppse try again')

# printiing an empty line for clear code or a little bit formatting
print()

if emp_sal_above_50_k:
    print('Employee who\'s salary is less or eqaul to 50K are =>')

    for key in emp_information:

        if emp_information[key]['Salary'] <= 50_000:
            print(f'{key} => {emp_information[key]}')

elif not emp_sal_above_50_k:
    print('No one has a Salary equal or less than 50K')

else:
    print('Awesome Sushant') #This will never print😢

# Output
```

```
# Employee Information dictionary is empty. If you want to add information then type yes[y] or type no[N] to leave
#  =>  y

# Enter empoyee Name
#  =>  SS
# Enter SS ID
#  =>  0
# Enter SS Salary
#  =>  999999
# Enter SS Mobile Number
#  =>  1234567890

# If you want to add more entries to the employee dictionary then type yes[y] or type no[N] to leave
#  =>  y

# Enter empoyee Name
#  =>  Jinex
# Enter Jinex ID
#  =>  1
# Enter Jinex Salary
#  =>  999999
# Enter Jinex Mobile Number
#  =>  1234567891

# If you want to add more entries to the employee dictionary then type yes[y] or type no[N] to leave
#  =>  n


# No one has a Salary equal or less than 50K
```

Practical 7

Write a Python Program to Create a dictionary of Library information (bookid, bookname, bookprice, bookauthor), input five values from the user and display the library, also change some values by searching the bookname in this dictionary.

Code

```
import json

lib = {}
```

```
is_sushant_awesome = 'well yes'

while is_sushant_awesome:

    # If dictionary is empty then this will run
    if not lib:
        # asking user to add information or leave
        yes_or_no = input("Library is empty. If you want to add information then type
yes[y] or type no[N] to leave\n ->  ").lower().strip()
        print() # empty line

    # If dictionary is not empty then this will run
    elif lib:
        yes_or_no = input("\nIf you want to add more entries to the Library then type
yes[y] or type no[N] to leave\n ->  ").lower().strip()
        print() # empty line

    else:
        print('Whoppse try again')

    # Breaking break
    if yes_or_no == 'no' or yes_or_no == 'n':
        break

    elif yes_or_no == 'yes' or yes_or_no == 'y':

        book_ID = input('Enter ID of book\n ->  ')
        book_name = input(f'Enter name of book for ID {book_ID}\n ->  ')
        book_author = input(f'Enter {book_name} author name\n ->  ')
        book_price = int(input(f'Enter {book_name} Price\n ->  '))

        # adding item to dict
        lib[book_ID] = dict([('Name',book_name), ('Author',book_author),
('Price',book_price)])

    else:
        print('Whoppse try again')

print(f"Library -> {json.dumps(lib, indent=4)}")

if lib:
    while True:
        yes_no = input('do you want to modify some books values. yes[Y] or no[N]\n -
>  ').strip().lower()
        if yes_no == 'yes' or yes_no == 'y':
```

```
            any_value_modified = False
            book_srch = input("Enter the ID of the book you want to Modify\n ->  ")

            if book_srch in lib:
                what_to_change = input("What you want to change\n - press 1 for
Name\n - press 2 for Author\n - press 3 for Price\n  ->  ").strip().lower()

                if what_to_change == '1' or what_to_change == 'name':
                    new_book_name = input(f'Enter new Name for book of ID
\'{book_srch}\'\n ->  ')
                    lib[book_srch]['Name'] = new_book_name
                    any_value_modified = True

                elif what_to_change == '2' or what_to_change == 'author':
                    new_book_author = input(f'Enter new Author name for book of ID
\'{book_srch}\'\n ->  ')
                    lib[book_srch]['Author'] = new_book_author
                    any_value_modified = True

                elif what_to_change == '3' or what_to_change == 'price':
                    new_book_price = input(f'Enter new Price for book of ID
\'{book_srch}\'\n ->  ')
                    lib[book_srch]['Price'] = new_book_price
                    any_value_modified = True

                else:
                    print('whoopse try again')

                if any_value_modified:
                    print(f"modified book with ID '{book_srch}' ->
{json.dumps(lib[book_srch], indent=4)}")

            else:
                print(f"book with ID '{book_srch}' doesn't exist in the book")

        elif yes_no == 'no' or yes_no == 'n':
            break
        else:
            print('whhooopppppsssseee didn\'t understand.')

else:
    print('Library is empty, Enter some books')

# Output
```

```
# Library is empty. If you want to add information then type yes[y] or type no[N] to
leave
#  ->  y

# Enter ID of book
#  ->  1
# Enter name of book for ID 1
#  ->  Awesome
# Enter Awesome author name
#  ->  SD
# Enter Awesome Price
#  ->  9876

# If you want to add more entries to the Library then type yes[y] or type no[N] to
leave
#  ->  n

# Library -> {
#     "1": {
#         "Name": "Awesome",
#         "Author": "SD",
#         "Price": 9876
#     }
# }
# do you want to modify some books values. yes[Y] or no[N]
#  ->  n
```

Practical 8

Write a Python Program to Create a hotel dictionary (bookingid, clientname, date, roomno) delete a value based on bookingid.

Code

```
import json

hotel = {}
is_sushant_awesome = 'well yes'

while is_sushant_awesome:

    # If dictionary is empty then this will run
    if not hotel:
        # asking user to add information or leave
```

```
        yes_or_no = input("Hotel is empty. If you want to add information then type
yes[y] or type no[N] to leave\n ->  ").lower().strip()
        print() # empty line

    # If dictionary is not empty then this will run
    elif hotel:
        yes_or_no = input("\nIf you want to add more entries to the Hotel then type
yes[y] or type no[N] to leave\n ->  ").lower().strip()
        print() # empty line

    else:
        print('Whoppse try again')

    # Breaking break
    if yes_or_no == 'no' or yes_or_no == 'n':
        break

    elif yes_or_no == 'yes' or yes_or_no == 'y':

        booking_ID = input('Enter booking ID\n ->  ')
        client_name = input(f'Enter Name of Client\n ->  ')
        booking_date = input(f'Enter the booking date\n ->  ')
        room_no = int(input(f'Enter {client_name} room number\n ->  '))

        # adding item to dict
        hotel[booking_ID] = dict([('Name',client_name), ('Date',booking_date),
('Room',room_no)])

    else:
        print('Whoppse try again')

print(f"Hotel -> {json.dumps(hotel, indent=4)}")

id_to_del = input('Enter the ID you want to delete\n ->  ')

if id_to_del in hotel:
    hotel.pop(id_to_del)
    print(f"booking detail with ID {id_to_del} has been deleted")

else:
    print(f"anything with ID {id_to_del} doesn't exit in the Hotel")

# Output
# Hotel is empty. If you want to add information then type yes[y] or type no[N] to
leave
```

```
#  ->  y

# Enter booking ID
#  ->  1
# Enter Name of Client
#  ->  SS
# Enter the booking date
#  ->  15/4/2022
# Enter SS room number
#  ->  2

# If you want to add more entries to the Hotel then type yes[y] or type no[N] to leave
#  ->  n

# Hotel -> {
#     "1": {
#         "Name": "SS",
#         "Date": "15/4/2022",
#         "Room": 2
#     }
# }
# Enter the ID you want to delete
#  ->  1
# booking detail with ID 1 has been deleted
```

Practical 9

Write a Python Program to implement python maths function/method (pi, e, sqrt, ceil,floor, pow, fabs, sin, cos, tan) using math module.

Code

```python
import math

# printing the value of PI
print(f"the value of pi is{math.pi}")

# printing Eulers number
print(f"the Euler constant is {math.e}")

# finding square root
print(f"the square root of 625 is {math.sqrt(625)}")

# round of by ceil and floor
```

```python
print(f"546.7856 will get convert to \"{math.ceil(546.7856)}\" by ceil function and
\"{math.floor(546.7856)}\" by floor function in the math module")

# pow function
print(f'5 raise to power 10 is {math.pow(5,10)}')

#Fabs function
print(f'-456546 will get convert to {math.fabs(-456546)} by fabs() function')

# sin function
print(f'the sine of 45 {math.sin(math.radians(45))}')

# cos function
print(f'the cosine of 45 {math.cos(math.radians(45))}')

# tan function
print(f'the tangent of 45 {math.tan(math.radians(45))}')

# Output
# the value of pi is3.141592653589793
# the Euler constant is 2.718281828459045
# the square root of 625 is 25.0
# 546.7856 will get convert to "547" by ceil function and "546" by floor function in
the math module
# 5 raise to power 10 is 9765625.0
# -456546 will get convert to 456546.0 by fabs() function
# the sine of 45 0.7071067811865476
# the cosine of 45 0.7071067811865476
# the tangent of 45 0.9999999999999999
```

Practical 10

Write a Python program to implement the random function random, randint, Randrange using random module.

Code

```python
import random

print('''
The random() method returns a random floating number between 0 and 1.
The randint() method returns an integer number selected element from the specified
range. This method is an alias for randrange(start, stop+1).
The randrange() method returns a randomly selected element from the specified range.
```

```
''')

# A game using functions from random module
print('    ####    <--> Snake(🐍) <-->    <--> Water(🌊) <-->    <--> Gun(🔫) <--
>     ####    ')
print()

awesome = "Sushant"

while awesome:
    choice = input("Choose from Snake[S], Water[W] and Gun[G] or type quit[Q] to
leave\n ->  ").strip().lower()
    comp_choice = ['s','w','g']
    random_num = random.randrange(0,3)
    comp_choice = comp_choice[random_num]

    if choice == 'quit' or choice == 'q':
        break

    # time when user will win
    elif (choice == 'snake' or choice == 's') and (comp_choice == 'w'):
        print('your choice -> Snake')
        print('computer\'s choice -> Water')
        print(' - you win - ')

    elif (choice == 'water' or choice == 'w') and (comp_choice == 'g'):
        print('your choice -> Water')
        print('computer\'s choice -> Gun')
        print(' - you win - ')

    elif (choice == 'gun' or choice == 'g') and (comp_choice == 's'):
        print('your choice -> Gun')
        print('computer\'s choice -> Snake')
        print(' - you win - ')

    # time when user will loose
    elif (choice == 'snake' or choice == 's') and (comp_choice == 'g'):
        print('your choice -> Snake')
        print('computer\'s choice -> Snake')
        print(' - you loose - ')

    elif (choice == 'water' or choice == 'w') and (comp_choice == 's'):
        print('your choice -> Water')
        print('computer\'s choice -> Snake')
        print(' - you loose - ')
```

```
    elif (choice == 'gun' or choice == 'g') and (comp_choice == 'w'):
        print('your choice -> Gun')
        print('computer\'s choice -> Water')
        print(' - you loose - ')

    else:
        print('WHOOPSE didn\'t understand')
    print()

# Output
# The random() method returns a random floating number between 0 and 1.
# The randint() method returns an integer number selected element from the specified
range. This method is an alias for randrange(start, stop+1).
# The randrange() method returns a randomly selected element from the specified range.

#     ####    <--> Snake(🐍) <-->    <--> Water(🌊) <-->    <--> Gun(🔫) <--
>     ####

# Choose from Snake[S], Water[W] and Gun[G] or type quit[Q] to leave
#  ->  w
# your choice -> Water
# computer's choice -> Snake
#  - you loose -

# Choose from Snake[S], Water[W] and Gun[G] or type quit[Q] to leave
#  ->  s
# your choice -> Snake
# computer's choice -> Water
#  - you win -

# Choose from Snake[S], Water[W] and Gun[G] or type quit[Q] to leave
#  ->  g
# your choice -> Gun
# computer's choice -> Snake
#  - you win -

# Choose from Snake[S], Water[W] and Gun[G] or type quit[Q] to leave
#  ->  q
```

Practical 11

Write a Python program to implement the statistics functions (mean, median, mode) using statistics module.

Code

```python
import statistics

sd = [5,5,5,6,7,3,5,6,2,3,4]

print(f"Mode of the list of number {sd} is {statistics.mode(sd)}")
print(f"Median of the list of number {sd} is {statistics.median(sd)}")
print(f"Mean of the list of number {sd} is {statistics.mean(sd)}")

# Output
# Mode of the list of number [5, 5, 5, 6, 7, 3, 5, 6, 2, 3, 4] is 5
# Median of the list of number [5, 5, 5, 6, 7, 3, 5, 6, 2, 3, 4] is 5
# Mean of the list of number [5, 5, 5, 6, 7, 3, 5, 6, 2, 3, 4] is 4.636363636363637
```