**Day1**

1. Input your name into a variable called $name and then print "Hello, <your name here>".

2. Write a program that adds two numbers and then prints out whether the sum of those two numbers is positive or negative.

3. Write a program that stores a number and keeps trying to get user input until the user enters the number correctly. As soon as the correct number is entered, it prints: Correct!

4. Input your first name and last name as two separate variables, labeled as $firstname and $lastname respectively. Concatenate them together using the dot operator '.' into a new variable called $wholename. Then print out the $wholename.

5. Write a program to accept an input string from the user and toggle the character cases. For example, $str=” Hello How Are You?”

O/p : hELLO hOW aRE yOU

6. Write a program which will perform sum and multiplication ,that sums and multiplies (respectively) all the numbers in a list of numbers. For example, sum([1, 2, 3, 4]) should return 10, and multiply([1, 2, 3, 4]) should return 24.

7. Write a program that takes a value (i.e. a number, string, etc) x and a list of values a, and returns True if x is a member of a, False otherwise. (Note that this is exactly what the in operator does, but for the sake of the exercise you should pretend Python did not have this operator.)

8. Write a program that has two lists and print True if they have at least one member in common, False otherwise. You may use your is\_member() function, or the in operator, but for the sake of the exercise, you should (also) write it using two nested for-loops.

9. Write a program for histogram that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following:

\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

10 . List of athletes participated in 100m and 200m running race

a. hundred\_meters = [‘Vikay’,’John’,’Kumar’,’Rajesh’,’Malar’,’Vaihai’]

b. two\_hundred\_meters = [‘Vetry’,’Petter’,’Priyanka’,’Kumar’,’Malar’]

Find the answer for below question from above lists (using set and Boolean Operators) a. Find the athletes who participated only in 100m race

b. Find the athletes who participated only in 200m race

c. Find the athletes who participated both 100m and 200m race

d. Find the athletes who participated only one race

11. Find the duplicate numbers from below list

a. List = [5, 8,4,18,8,55,6,8,3,18,5,3,44,]

Internal Use - Confidential

12. Reverse the below list without using any inbuilt keywords (like reverse() or [::-1]) List = ["cat","tiger","lion", "zebra" , "crocodile", "snack"]

**Day2**

1. Define a function generate\_n\_chars() that takes an integer n and a character c and returns a string, n characters long, consisting only of c:s. For example, generate\_n\_chars(5,"x") should return the string "xxxxx". (Python is unusual in that you can actually write an expression 5 \* "x" that will evaluate to "xxxxx". For the sake of the exercise you should ignore that the problem can be solved in this manner.)

2. The function max() from exercise 1) and the function max\_of\_three() from exercise 2) will only work for two and three numbers, respectively. But suppose we have a much larger number of numbers, or suppose we cannot tell in advance how many they are? Write a function max\_in\_list() that takes a list of numbers and returns the largest one.

3. Write a program that maps a list of words into a list of integers representing the lengths of the correponding words.

4. Write a function find\_longest\_word() that takes a list of words and returns the length of the longest one. Modify the same to do with lambda expression.

5. Write a function filter\_long\_words() that takes a list of words and an integer n and returns the list of words that are longer than n. Modify the same to do with lambda expression.

6. Write a version of a palindrome recognizer that also accepts phrase palindromes such as "Go hang a salami I'm a lasagna hog.", "Was it a rat I saw?", "Step on no pets", "Sit on a potato pan, Otis", "Lisa Bonet ate no basil", "Satan, oscillate my metallic sonatas", "I roamed under it as a tired nude Maori", "Rise to vote sir", or the exclamation "Dammit, I'm mad!". Note that punctuation, capitalization, and spacing are usually ignored.

7. A pangram is a sentence that contains all the letters of the English alphabet at least once, for example: The quick brown fox jumps over the lazy dog. Your task here is to write a function to check a sentence to see if it is a pangram or not.

8. Represent a small bilingual lexicon as a Python dictionary in the following fashion {"merry":"god", "christmas":"jul", "and":"och", "happy":gott", "new":"nytt", "year":"år"} and use it to translate your Christmas cards from English into Swedish. That is, write a function translate() that takes a list of English words and returns a list of Swedish words.

9. Write a function char\_freq() that takes a string and builds a frequency listing of the characters contained in it. Represent the frequency listing as a Python dictionary. Try it with something likechar\_freq("abbabcbdbabdbdbabababcbcbab").

10. Create a module called mathematics.py and provide subroutines (should be defined generally and should work for any number of arguments) such as:

Add Sub

Sort the values Max Sort

Internal Use - Confidential

Use the module in a program and apply the functions on two array variables (say a and b) to: Add two arrays (and store it in c) Subtract two arrays(and store it in d)

Find the minimum and maximum value of the resultant array (c or d) Sort the resultant array(c or d) 11. Try above programme with package.

**Day3**

1. Create a Date class, which represents the Date with its attributes. Write a UseDate class, which makes use of the Date class to instantiate, and call methods on the object.

2. WAP to read data from one file and writes in second file.

3. WAP which will display different function of math and numpy library.

4. Get name, id number and location from user and print it in a single line (use format method)

5. Swap the last 2 octet of the IP address (use split and join)

a. Eg . input : 172.65.234.45 output : 172.65.45.234

6. Reverse the below list without using any inbuilt keywords (don’t use reverse() or [::-1]) b. List = ["cat","tiger","lion", "zebra" , "crocodile", "snack"]

7. Get the employee name and rating (rating is 1 to 10) from the user like below and print the name and performance, use if conditions and try, except

Sample input : Raja : 6.5

Ask for valid input from user for below invalid data Rating is NULL

Rating contain characters Name is NULL

Name contain integers All are empty

8. Find the even numbers from below list (use list comprehension)

Internal Use - Confidential c. List = [5, 21,4,18,2,55,6,8,3,18,5,3,44,]

9. Create your own area module which should have 2 library files each should contain 2 functions (total 4 functions) , run and find the area of below

d. Rectangle

e. Square

f. Circle

g. Triangle ((height \* base)/ 2)

10. Sort the below list, don’t use any inbuilt functions

h. List = [6,2,8,9,34,12,4,3,9]

Internal Use - Confidential

11. Print the lines in the file, which contain the website address, create new directory, create new file save the output to new file, print the new file output and delete the old file

12. Write a program to get Fibonacci series for given number

13. Create student class which will take the name of the student as input to class, get the 5 subjects marks, print the total mark along with the name , if the student get less than 50% mark remove the student from class , finally print the total pass students

14. Get the employee name and rating (rating is 1 to 10) from the user like below and print the name and performance, use if conditions and try, except

Sample input : Raja : 6.5

Ask for valid input from user for below invalid data

1. Rating is NULL

2. Rating contain characters

3. Name is NULL

4. Name contain integers

5. All are empty

15. Print the lines in the file, which contain the website address, create new directory, create new file save the output to new file, print the new file output and delete the old file

16. Find the size of the folder, if its exceed 2Mb delete few files from the folder to bring the folder size to less than 2Mb

17. Get mark form user, mark should be in the range of 0 to 100, if user gived invalid mark rasie the assertion error

18. Log the user data in to file using logging module

**Day4**

1. Assume below list having temperature in Celsius convert in to Fahrenheit (use lambda function with map function)

i. Amount = [21,23,18,22,19,34,24,35]

2. Remove the out of sequence number in below list (sequence is 25 to 40)

3. [25, 28,26,4000,27, 31, 5,28,29,30,34,8,32,33, 35, 32,37,36, 40,38,39, 95] 4. Remove duplicate elements in the list , don’t use “set” function write your own code j. List = [3,5,2,6,8,3,18,5,3,44,21,4,18,2]

5. Program should add value to list, if the same number given again don’t add to list (don’t use set)

6. Write a program to find given string is Palindrome or Not (write a function pass the string to function , and get return True if string is Palindrome, return False if string is not Palindrome )

7. Write the python program for find the line contains MAC address in the given multiline string *%DAEMON-3-SNMPD\_TRAP\_WARM\_START: trap\_generate\_warm: SNMP 0E-2B-56-1C-89-*

Internal Use - Confidential

*0 trap: warm start*

*Feb 22 20:35:07 router1 snmpd[359]:*

*%DAEMON-6-SNMPD\_THROTTLE\_QUEUE\_DRAINED: trap\_throttle\_timer\_handler: cleared all throttled traps*

*Feb 23 07:34:56 3C-21-5A-14-81-0A router1 snmpd[359]: %DAEMON-3- SNMPD\_TRAP\_WARM\_START: trap\_generate\_warm: SNMP trap: warm start Feb 23 07:38:19 router1 snmpd[359]: %DAEMON-2-SNMPD\_TRAP\_COLD\_START: trap\_generate\_cold: SNMP trap: cold start*

8. Write the python program for find the line contains IP address in the given multiline string %DAEMON-3-SNMPD\_TRAP\_WARM\_START: trap\_generate\_warm: 253.67.34.9 SNMP

trap: warm start

Feb 22 20:35:07 router1 snmpd[359]: %DAEMON-6-

SNMPD\_THROTTLE\_QUEUE\_DRAINED: trap\_throttle\_timer\_handler: cleared all

throttled traps

Feb 23 07:34:56 router1 snmpd[359]: %DAEMON-3-SNMPD\_TRAP\_WARM\_START: trap\_generate\_warm: 10.145.78.2 SNMP trap: warm

start

Feb 23 07:38:19 router1 snmpd[359]: %DAEMON-2-SNMPD\_TRAP\_COLD\_START: trap\_generate\_cold: SNMP trap: cold start

Internal Use - Confidential