DRAFT SCHEME

Scheme of Valuation/Answer Key

(Scheme of evaluation (marks in brackets) and answers of problems/key)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, JUNE 2022

Course Code: CST362

Course Name: PROGRAMMING IN PYTHON

Max. Marks: 100		Duration:

PART A

Answer all questions, each carries 3 marks.

1		The '//' operator in Python returns the integer part of the floating number'/' normal division (a) 4 (b) 4.5 1.5 marks each

2		num = int (input ("Enter any number to test whether it is odd or even: ")
		read n numbers 1 mark
		for loop
		if (num % 2) == 0: 1 mark
		count_even incremented
		else:
		count_odd incremented correct program 1 mark

3		Use of negative indices is to refer to the end of the sequence: index value of -1 gives the last element, and -2 gives the second last element of an array. 1 marks Example , b = "Hello, World!" print(b[-5:-2]) output is orl 2 marks
4		of = open ("code. Txt" , " w ") 1 marks of. write ("PROGRAMMING IN PYTHON") 1 marks if = open (" code. txt " , " r ") text = if . read () print text 1 marks
5		SciPy, Scikit-Image Mahotas.,Pillow. OpenCV. SimpleITK Matplotlib. NumPy. any three 1 mark each
6		Import the Tkinter module. import Tkinter Create the GUI application main window. top = Tkinter.Tk() Add one or more of the above-mentioned widgets to the GUI application.

DRAFT SCHEME

		Enter the main event loop to take action against each event triggered by the user. top.mainloop() Three main steps 3 marks

7		def __del__(self):
		# body of destructor 1.5 marks
		obj = student()
		del obj 1.5 marks

8		Creating class and methods 1.5 marks Conversion method 1.5 marks
9		import random. 1.5 marks value = randint(1,10) 1.5 marks #1 and 10 represent the range for your random value. print(value)

10		The OS module in Python provides functions for creating and removing a directory (folder), fetching its contents, changing and identifying the current directory, etc. import the os module to interact with the underlying operating system . 1.5 marks
		import os
		# Get the current working directory (CWD)
		cwd = os.getcwd() 1.5 marks

		PART B
		Answer one full question from each module, each carries 14 marks.
		Module I

11	a)	n = int(input("Enter the value of n : "))
		for i in range(1,n+1):
		for j in range(1, i+1):
		print(j,end=" ")
		print()

	b)	Loop statements - While Loop and for loop, explanation with example 4 marks control statements continue, break and pass explanation with example 4 marks
		OR

12	a)	for i in range(1,1000):
		prime = True
		j = 2
		while (j <= i/2):

DRAFT SCHEME

		if (i % j == 0):
		prime = False
		break
		j = j + 1
		if (prime):
		print (i, end = ' ')

	b)	x1=int(input("enter x1 : ")) x2=int(input("enter x2 : ")) y1=int(input("enter y1 : ")) y2=int(input("enter y2 : ")) result= ((((x2 - x1)**2) + ((y2-y1)**2))**0.5) 3 marks for equation + 3 marks for correct output
		Module II

13	a)	strr = "Given string"
		dict = {}
		for i in strr:
		if i in dict:
		dict[i] += 1
		else:
		dict[i] = 1
		print ("Count of all characters :\n " + str(dict))
		Output
		Count of all characters :
		{'G': 1, 'i': 2, 'v': 1, 'e': 1, 'n': 2, ' ': 1, 's': 1, 't': 1, 'r': 1, 'g': 1} 5 marks correct program 2 marks explanation

	b)	def min_max(list_elements): Find smallest number Find largest number return smallest, largest 5 marks correct program 2 marks explanation
		OR
14	a)	n = int(input("Enter how many numbers you want to enter: ")) arr=[]

DRAFT SCHEME

		positive=[]
		negative=[]
		for i in range(n):
		number=int(input())
		arr.append(number)
		for i in range(n):
		if arr[i]>=0 :
		positive.append(arr[i])
		else:
		negative.append(arr[i]) 5 marks correct program 2 marks explanation

	b)	Birthdays={'A':'03/14/1879','B':'03/14/1879','C':'03/14/1879','D':'06/14/18669',} print('Who's birthday do you want to look up?') In_name = input() print(Birthdays[In_name]) 5marks correct program 2 marks explanation
		Module III
15	a)	import turtle 1 mark ws = turtle.Screen() 1 mark geekyTurtle = turtle.Turtle() 1 mark for i in range(5): geekyTurtle.forward(100) geekyTurtle.right(144) 2 marks
	b)	Any four operations open, read, save, rotate, crop, thumbnail etc…with python code 2 marks each 1 marks for image processing library 1 marks
		OR
16	a)	Two text box , labels one button 3 marks each Conversion equation 1 mark
	b)	from tkinter import * 1 mark %from PIL import ImageTk,Image /// or using PIL root = Tk() 1 mark canvas = Canvas(root, width = 300, height = 300) canvas.pack()

DRAFT SCHEME

		img = PhotoImage(file="image.jpg") 3 marks % img = ImageTk.PhotoImage(Image.open("image.jpg")) ////or using pil canvas.create_image(20,20, anchor=NW, image=img) 2 marks mainloop() 1 mark
		Module IV
17	a)	polymorphism – 2 marks function overloading 3 marks Suitable example. 2 marks
	b)	multi-level inheritance explanation 3 marks program 4 marks
		OR
18	a)	Abstract Base Class explanation 1 mark Definition of two class - 2 marks Correct program - 4 marks
	b)	In Python, exceptions can be handled using a try statement. The critical operation which can raise an exception is placed inside the try clause. The code that handles the exceptions is written in the except clause. 3 marks Example program – 4 marks
		Module V
19	a)	numpy arrays - 2 marks matrix multiplication program – 5 marks
	b)	Matplotlib plot 2 marks suitable legends, 2 marks labels 2 marks and a title. 1mark
		OR
20	a)	import pandas as pd df = pd.read_csv('employee.csv") 3 marks 1) df.head(7) 2) df= df.sort_values('name',asvending=True) print (df) 3) df=df[['name','salary']][df.salary==df['salary'].max()] 4) df=df[['name']][df.gender=='m'] 5) df=len(pd.unique(df['team']))

DRAFT SCHEME

		1 marks for each 5 marks
	b)	import pandas as pd 2 marks df = pd.DataFrame(insert values…….. 2 marks columns=['Reg_no ','Name','Sub_Mark1',' Sub_Mark2",' Sub_Mark3']) 2 marks print("data frame =",df)
		****