Register Number									

Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam – 603 110 (An Autonomous Institution, Affiliated to Anna University, Chennai)						
Computer Science and Engineering **Continuous Assessment Test -2** **Question Paper**						
Degree & Branch	B.E				**Semester**	VII
Subject Code & Name	UCS1727– GPU Computing				**Regulation:**	**2018**
Academic Year	2023-2024 ODD	**Batch**	2020-2024	**Date**	**18.10.2023**	**FN**
Time: 08:10 - 09:40 AM (90 Minutes)	**Answer All Questions**				**Maximum: 50 Marks**	

(K1: Remembering, K2: Understanding, K3: Applying, K4: Analyzing, K5: Evaluating)

CO1:	Understand GPU architecture (K2)
CO2:	Write programs using CUDA, identify issues and debug them (K3)
CO3:	Implement efficient algorithms in GPUs for common application kernels such as matrix multiplication (K3)
CO4:	Write simple programs using OpenCL (K3)
CO5:	Write an efficient parallel program for a given problem(K3).

Part – A (6×2 = 12 Marks)

1	K1	What are eager evaluation and lazy evaluation?	CO2	2.1.2 2.2.2 13.1.1
2	K1	What is speculative execution?	CO2	1.4.1 2.1.2 2.2.2
3	K3	Identify the CUDA API call used to retrieve GPU device properties?.	CO2	2.1.2 13.3.2
4	K1	What is loop invariant analysis?.	CO2	2.1.2 2.2.2
5	K1	What does the term "cache coherence" refer to, and what are the available approaches to address cache coherence?	CO3	2.2.2 13.1.1 13.1.2
6	K3	A CUDA kernel performs the following operation C[z] = A[y] * B[x] Identify whether the kernel is memory bound or arithmetic bound? Justify your answer.	CO3	1.4.1 2.1.2 13.3.1

Part – B (3×6 = 18 Marks)

7	K3	Explain the role and application of CUDA ballots in GPU programming. Develop a code snippet in which CUDA ballots prove advantageous for coordinating threads. .	CO2	1.4.1 2.2.2 13.1.2 13.3.1

8	K3	Apply the concept of loop fusion for the following code snippet and demonstrate how it improves performance.
		unsigned int i,j;
		a = 0;
		for (i=0; i<100; i++)
		{
		a += b * c * i;
		}
		d = 0;
		for (j=0; j<200; j++)
		{
		d+= e * f
		}
			CO2	1.4.1
				2.2.2
				13.1.1

9	K3	Apply the concept of loop unrolling for the following loop structure demonstrate how loop unrolling will improve performance in parallel programming. for (i=1; i<=1000; i++) X[i] = X[i] + S; X is an array whose starting address is stored in the register R1 R2 contains the terminal address of an array x S is a constant stored in the register F2	CO2	1.3.1 1.4.1 **2.2.2** 13.3.1

Part – C (2×10 = 20 Marks)

10	K2	Explain the approach for error handling in CUDA programming.	CO3	1.3.1 1.4.1 2.2.2 13.3.1 13.3.2
		(OR)		
11	K2	Describe the challenges related to algorithmic aspects in CUDA programming.	CO3	1.3.1 1.4.1 2.2.2 13.3.1 13.3.2
12	K3	Create a CUDA program that employs the Binary Search technique to locate an element within an array.	CO2	1.4.1 2.1.2 2.2.2 13.3.1 13.3.2
		(OR)		
13	K3	Develop a CUDA program that populates an array with values ranging from 0 to num_elements. Additionally, set up four streams to run concurrently on four different GPU devices. The objective is to measure the following metrics for each of the four GPU devices: • The duration it takes to transfer data from the CPU to the GPU. • The time required to execute the kernel operation. • The time it takes to copy the results back from the GPU to the CPU • The total execution time of the entire operation	CO2	1.4.1 2.1.2 2.2.2 13.3.1 13.3.2