**Department of Computing**

**CS-361: Computer Graphics  
Class: BSCS-12ABC & SE12AB**

**Lab 10: Fluid, Cloth, and Particle Simulations  
 [Open Ended Lab]**

**CLO 2 -** Apply mathematical and algorithmic principles to implement basic computer graphics techniques, such as line drawing and shading.

**CLO 3-** Develop interactive graphics applications using modern graphics APIs such as OpenGL or DirectX.

**CLO 4 -** Design and implement 2D and 3D graphical solutions for real-world problems.

**Date: 15th April 2025**

**Time: 2:00 PM – 4:50 PM**

# Instructor: Dr. Sidra Sutana

# Lab Engineer: Mr. Aftab Farooq

**Name:** Ahmed Mohiuddin Shah

**CMS ID:** 415216

**Section:** BSCS-12-A

**Lab:** 10

**Lab 10: Fluid, Cloth, and Particle Simulations – [Open Ended Lab]**

**Introduction:**This is an open-ended lab focusing on implementing and experimenting with different types of simulations in computer graphics. Students will choose different simulation types to implement and explore.

### **Lab Objective:**

Design and develop an interactive simulation environment that demonstrates below following systems:

* Fluid Simulation (Grid-based or Particle-based)
* Cloth Simulation (Mass-Spring System)
* Particle Systems (e.g., Fire, Smoke, Rain)

## Tools/Software Requirement:

* **Operating System:**
  + Windows / macOS / Linux (Ubuntu recommended)
* **Development Environment:**
  + **Windows:** [Code::Blocks](http://www.codeblocks.org/) or [Visual Studio](https://visualstudio.microsoft.com/)
  + **macOS:** [Xcode](https://developer.apple.com/xcode/)
  + **Linux:** GCC and g++ compilers
* **Graphics Libraries:**
  + **OpenGL** (built-in on macOS and Linux, available in Windows IDEs)
  + **GLUT** (OpenGL Utility Toolkit)
  + **GLEW** (OpenGL Extension Wrangler Library)
* **Package Manager (for macOS/Linux):**
  + **Homebrew** (macOS): brew install freeglut glew
  + **APT** (Linux): sudo apt-get install freeglut3-dev glew-utils
* **Compilers:**
  + **Windows:** MinGW (for Code::Blocks) or Microsoft C++ Compiler (for Visual Studio)
  + **macOS/Linux:** GCC/G++
* A programming environment (e.g., Visual Studio, PyCharm, or any IDE of your choice).
* A graphics library (optional, e.g., OpenGL, SDL, or a simple image library for saving images).
* Basic knowledge of simulations.

### **Task Description:**

You are to create a physics-based simulation sandbox where users can interact with the scene in real-time. Your project should include:

**Core Requirements:**

* Implementation of the following:  
  ▸ Fluid dynamics (e.g., splash, flow, or smoke)  
  ▸ Cloth behavior (e.g., waving in wind, draping over obstacles)  
  ▸ Particle effects (e.g., explosion, fire, rainfall)
* User interaction:  
  ▸ Allow input via mouse/keyboard to influence simulations (e.g., apply force, move an object, inject particles)
* Visualization:  
  ▸ Render particles, fluids, and/or cloth using appropriate visual cues  
  ▸ Real-time performance and clear display of forces/effects

**Expected Feature :**

* Collision detection between cloth and other simulated elements
* Simulate fluid affecting cloth or interacting with particle systems
* Add GUI controls to toggle parameters (e.g., gravity, wind, viscosity)

### **Deliverables:**

 Compile a single word document by filling in the solution part and submit this Word file on LMS

 Include screenshots of the program outputs.

 Submit your Lab Word File and code files seperately on submission link.

# Lab Rubrics

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Lab Rubrics for (Lab-10:Fluid, Cloth, and Particle Simulations – **[Open Ended Lab]**) | | | | | |
|  | | | | | |
| **Sr.**  **No.** | **Assessment** | **Unacceptable (0 Marks)** | **Does Not Meet Expectations (1/2 Marks)** | **Meets Expectations (3/4 Marks)** | **Exceeds Expectations (5 Marks)** |
| **1** | **Illustrating the basic understanding of semantics and syntax**  **(CLO3, PLO5)** | The student did not submit any work.  OR  The student plagiarized the solution and/or used unfair means. | The student is unable to demonstrate the understanding of syntax of C language and is unable to write an executable code.  The student is not able to understand the structure of a program at all. | The student demonstrates some understanding of syntax of C language and is able to write a code with few errors.  The student is able to understand the structure but still learning the syntax. | The student demonstrates good understanding of syntax of C language and is able to write executable code without help  The student is able to understand the structure and is able to identify problems in the code  when introduced |
| **2** | **Software Tool Usage**  **(CLO4-PLO3)** | The student demonstrates a lack of understanding of tool usage.  Implementation has syntax/semantic/runtime errors, and the student is unable to debug and correct the errors.  The code has inadequate comments and variable names and does not adhere to the coding standards.  No Error handling has been performed.  Documentation is poorly structured. | The student demonstrates some understanding of tool usage.  The codes are correct in terms of their syntax, however, the program output is not always correct in all test cases.  The code has limited comments and inconsistent variable names and may not adhere to the coding standards.  Some Error handling has been performed.  Documentation is adequately structured. | The student demonstrates a good understanding of tool usage.  Furthermore, his/her coding is complete and functional, and the program output is correct in all test cases.  The code has sufficient comments and consistent variable names and reasonably adhere to the coding standards.  Adequate Error handling has been performed.  Documentation is well structured. |