SPL1 Project Proposal Form, 2025
Institute of Information Technology (IIT)
University of Dhaka

Student's Name:	A. M. Toufiq Alam		
Student's Roll:	**43**	**Phone:**	**01540405402**

Project Description:

A implementation of the classic Pac-Man arcade game developed in C++ utilizing the SFML (Simple and Fast Multimedia Library) framework. This project demonstrates advanced programming concepts including artificial intelligence algorithms, data structure optimization, and real-time graphics rendering.

Technical Architecture

Core Technologies

- **Programming Language**: C++
- **Graphics Framework**: SFML 3.x

System Architecture

Modular design with clear separation of concerns:

- **Game Logic Layer**: State management, collision detection, game rules
- **AI Processing Layer**: Pathfinding algorithms for autonomous agents
- **Rendering Layer**: Visual output and animation systems
- **Data Persistence Layer**: File I/O for maze configurations and score management

Algorithm Implementation

1. A* Pathfinding Algorithm

Enables intelligent ghost navigation through maze environment.

Key Features:

- Manhattan distance heuristic for grid-based pathfinding
- Custom min-heap implementation for priority queue

- Hash maps for O(1) closed set lookups
- Dynamic memory management with proper cleanup

2. Merge Sort Algorithm

Efficient score sorting for leaderboard generation.

Features:

- Recursive divide-and-conquer approach
- Stable sorting maintaining relative order
- O(n log n) time complexity

Data Structures

1. **2D Vector Matrix**: Dynamic maze representation with flexible sizing
2. **Hash Maps**: Efficient node tracking in pathfinding (O(1) average lookup)
3. **Custom Min-Heap**: Priority queue for A* algorithm (O(log n) operations)
4. **Arrays**: Fixed-size collections for ghost positions and hazard zones

Advanced Features

Dynamic Difficulty Scaling

Three progressive levels with parameterized adjustments:

- Ghost population: 2, 4, 6 entities
- Hazard zones: 1, 3, 5 zones
- Variable AI movement frequencies

File I/O Management

- Text-based maze configuration with character mapping
- Persistent score system maintaining top 100 scores per difficulty
- Dynamic parsing supporting arbitrary maze dimensions

Languages or Tools to be used:

Programming Language

- **C++** (C++11 or later)
  - Standard Template Library (STL)

Graphics Library

- **SFML 3.x** (Simple and Fast Multimedia Library)
  - `SFML/Graphics.hpp` - Graphics rendering, shapes, text
  - Window management and event handling
  - Real-time rendering capabilities

Development Tools

- **C++ Compiler** (GCC, Clang, or MSVC)
- **Build System** (Make, CMake, or IDE-specific)
- **Text Editor/IDE** (VS Code,Xcode etc.)

Operating System

- **macOS** (based on font paths, but adaptable to Windows/Linux)

Supervisor's Name:________________________________

Signature of the supervisor:_________________________

Date: ________________________

Before the Midterm Presentation:

I confirm that the progress is satisfactory and I am forwarding it for midterm presentation.

Signature of the supervisor:_________________________

Date: ________________________

Midterm Presentation Feedback: