Computer Organization &

Assembly Language

Fall 2023

Final Project

"PacMan Game"

Deadline: *December 8th, 2023*

Instructions

- You are required to work in MASM615. Project incompatible of MASM615 will not be considered.
- User Interface is important in this project. Try to develop an attractive user interface.
- Use of extra features in the project is encouraged.
- Use good programming practices (well commented and indented code, meaningful variable names, readable code etc.).
- Only one group of the student submit the project in Zip File.

  FORMAT: RollNo.zip.
- Evaluation Criteria will be shared with you soon.
- **You need to submit video of your full game**
- **You can use Irvine library only for inputs and graphics other than that you will be awarded with zero marks.**
- You are not allowed to use directives (e.g., if else, while)
- Copy/cheating and code copy from any ai Source case will be awarded an “F” grade in the course.

Introduction

The goal of this project is to implement a classic Pac-Man game using Assembly language (8086) in Visual Studio. The game will consist of three levels, each with unique features, challenges, and objectives. Students will need to demonstrate their understanding of Assembly language programming, game logic, and graphics manipulation.
Players embark on their Pac-Man journey with three lives. Losing a life occurs when Pac-Man collides with a ghost or faces specific challenges within the maze. The inclusion of lives adds an extra layer of strategy, emphasizing the importance of careful navigation and strategic use of power pellets. You are required to draw the levels and the full functionality for the pac man. It includes the movement of the player the movement of ghost i.e. ai movement. The wall is around the ground it is shown in images below.

LEVEL

Level 1: The Beginning

1. Maze Layout:

The first level introduces players to a straightforward maze layout characterized by simple paths and walls. The maze is designed to be easily navigable, allowing players to familiarize themselves with Pac-Man's movement dynamics.

2. Ghost Behavior:

Ghosts in this level exhibit basic behavior, primarily focusing on chasing Pac-Man. Their movements are predictable, providing players with a manageable introduction to ghost interactions.

3. Objective:

The primary goal of Level 1 is for Pac-Man to collect all the dots scattered throughout the maze. Unlike subsequent levels, there are no power pellets introduced at this stage, emphasizing the fundamental mechanics of dot collection.

4. Lives:

Players begin their Pac-Man journey with three lives. Losing a life occurs when Pac-Man collides with a ghost.

5. Level Design Rationale:

Level 1 serves as an introductory phase, allowing players to grasp the fundamental mechanics of the game. The straightforward maze layout and basic ghost behavior provide a gentle learning curve, setting the foundation for more complex challenges in the following levels.

Level 2: The Challenge

6. Maze Layout:

The second level ramps up the complexity with additional walls, paths, and strategically placed fruit bonuses. The maze design requires players to navigate more carefully and consider new challenges.

7. Ghost Behavior:

Ghosts in Level 2 exhibit more diverse behaviors, introducing the concept of cutting off Pac-Man's path. Pinky, a new ghost, adds complexity by employing unpredictable movements and ambushing strategies.

8. Objective:

In addition to collecting dots, Pac-Man must strategically eat fruit bonuses for extra points. Power pellets are introduced, providing Pac-Man with the ability to temporarily eat ghosts.

9. Lives:

Players still start with three lives. Colliding with a ghost or facing specific challenges within the maze results in the loss of a life.

10. Level Design Rationale:

Level 2 challenges players with a more intricate maze layout and varied ghost behaviors. The introduction of power pellets adds a strategic layer to the game, requiring players to plan their movements to maximize points. The inclusion of three lives continues to emphasize the importance of careful navigation. In this Level there will addon ball upon eating which player will be able to kill ghost and add some points on each killing and ghost revert back to its initial point. Remember that these points are for some time i.e., when player eat that addon he/she will be able to kill only for 20 secs.

Level 3: The Showdown

11. Maze Layout:

Level 3 reaches the pinnacle of complexity, featuring teleportation paths, hidden shortcuts, and additional obstacles. The maze design demands advanced navigation skills from players.

12. Ghost Behavior:

Ghosts in Level 3 exhibit diverse and coordinated behaviors, making it significantly more challenging for Pac-Man. Inky and Clyde, new ghosts, introduce unique and complex strategies that require strategic planning from the player.

13. *Objective:*

Pac-Man faces the ultimate challenge, navigating a complex maze while collecting dots, eating fruit, and avoiding advanced ghost behaviors. Higher point values are assigned for collecting dots and eating ghosts. The level culminates in a showdown with a boss ghost with unique behavior.

14. *Lives:*

Players still start with three lives. Losing a life occurs when Pac-Man collides with a ghost or faces challenges within the maze.

15. *Level Design Rationale:*

Level 3 serves as the climax of the game, pushing players to their limits with intricate maze layouts and advanced ghost behaviors. The introduction of a boss ghost adds a unique twist, requiring players to adapt their strategies. The continued inclusion of three lives emphasizes the ongoing challenge of avoiding ghosts and completing the game. In this level there will be like cherry upon eating that we will be able to increase the player live by one.

File Handling

You are required to store all the score of all the players that have played the game in **sorted way**. For this you will use file handling. You need to store the name and updated highest score in a file.

Important Rules

- The walls you created in the picture below should be on your roll number i.e., suppose your roll number is 2583 so you go to only five i.e., 2553 so in the above there is 2 block then five the again five and then 3
- The screen size is 640x480.
- You should set the third level zombies/ghost to be more faster
- This is an individual project and you will be evaluated based on the demonstration that you must give before 08th December 2023.
- You must make sure that all the exceptions have been handled (e.g. Input provided in wrong format is not processed.
- You must use the procedures and the parameters are passed through the push and pop in stack.

You are required to use sound features for your game.

Screens

The following screens are given as an example. You can create your own screens which would include:

- Welcome Screen that takes your name input
- Game Menu screen
- Main Screen for GamePlay
- Pause Screen/Functionality
- Instructions Screen
- High Score and Players Names Display

Code Template

```
The Libray here
Like include lib here
The main code for assembly
The prototype types for file handling
Like CreateFileA Proto, a1:ptr byte, a2: dword, a3: Dword, a4: dword, a5: dword,
a6: dword, a7: dword
ReadFile PROTO, a1:DWORD, a2: PTR BYTE, a3: Dword, a4: ptr dword, a5: dword
WriteFile PROTO, a1:DWORD, a2: PTR BYTE, a3: Dword, a4: ptr dword, a5: dword
CloseHandle proto, a1:dword
Then there will be procs like
MainMenu Proc
ret
MainMenu ENDP

Level1 Proc
```

```
ret
Level1 ENDP

Level2 Proc
ret
Level2 ENDP

Level3 Proc
ret
Level3 ENDP

DrawPlayer Proc
Ret
DrawPlayer EndP

RandomPlayerPosition PROC
Ret
RandomPlayerPosition ENDP

DrawGhost Proc
Ret
DrawGhost ENDP

DrawEatables Proc
Ret
DrawEatables ENDP


Main Proc
Main ENDP
```

Main Screen for GamePlay

Above is the example representation how the game should look like the blue lines represent the wall. The “X” represents the player and dot represent the eatable. You can use these types of graphics in your game.

To better understand the game please fallow the link below

https://freepacman.org/