e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

CLASSIC PACMAN GAME

Dr. Seema Biday[*1], Yash Sanap[*2], Yukta Khadakban[*3], Anushka Alshi[*4], Sandesh Patil[*5]

[*1]HOD, Department Of Computer Engineering Terna Engineering College Navi Mumbai, India.

[*2,3,4,5]Student, Department Of Computer Engineering Terna Engineering College Navi Mumbai, India.

ABSTRACT

This paper is about implementing Classic PacMan games in python with the help of pygames library and search algorithms. Pac-Man is a 1980 maze video game created and released by Namco which was developed for arcades. The original Japanese title of Puck Man has been changed to Pac-Man for international release as a precautionary measure against the destruction of arcade machines by converting P into F. The player controls Pac-Man, who must eat all the dots inside the closed maze while avoiding the four colored ghosts.The main aim is to build an intelligent pacman game in which the spirits/ ghosts which are present in the maze finds optimal paths through the maze to find a particular goal such as a finding shortest distance from current position of spirit to the position where pacman i.e player is heading in the maze .For that, we have implemented AI search algorithms like A*search,Uniform cost search. For implementing the project we have made use of the pygame library of python which is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used with the Python programming language. This paper describes the technical details and implementation of a Classic PacMan game . A small demo example and python modules which are created during creating a game are presented in the implementation section, to show the implementation of the project.

Keywords: Pac Man; Maze; A*Search;Uniform Cost Search.

I. INTRODUCTION

Game development began in early 1979, under the direction of Toru Iwatani and a team of nine men. Iwatani wanted to make a play that would not attract women and men, because most video games at that time had military or sports themes. Although the Pac-Man character's inspiration was the Pac-Man image. sliced pizza, Iwatani said he also incorporated a Japanese word meaning mouth, chuchi. The characters inside the game are made beautiful and colorful to attract younger players. Puckman's first Japanese title was taken from the hockey-puck form of the main character.

Pac-Man is an action maze chase video game; the player controls the eponymous character through an enclosed maze. The objective of the game is to eat all of the dots placed in the maze while avoiding four colored ghosts or spirits that pursue him. When Pac-Man eats all of the dots, the player advances to the next level. If Pac-Man makes contact with a ghost, he will lose a life; the game ends .But if pacman eats up all the dots in the maze the player is the winner.

Pac-man is known to be one of the most popular arcade games in the world. The inspiration for this project comes from our love of video games. Pacman Game in python is a simple console or small clip game designed for entertainment purposes. Pacman must be driven in such a way as to navigate the previously defined blue path in order to clear or be eaten by Pacman. When you clear a path, the more points you get; the game is easy to play.

The game will be designed in python programming language The player must control the pacman of the arrow keys on his keyboard. If the Pac-man eats large dots it will turn the ghost blue which allows Pac-man to gain bonus points by eating them. To win the level, Pacman must eat all the food in the maze. Pacman will eat the food when he is done with it. If the Pac-man is captured by a ghost, then the game gets over.

The inspiration for choosing the PAC MAN game comes from our passion for playing video games. In recent years video games have become quite popular. Which have improved the gaming industry. Pac man games will help young children improve their concentration and concentration levels in a fun way.

Pygame Library is an open source software module for Python program designed to help you play games with other multimedia applications. Built on the most portable SDL (Simple DirectMedia Layer) development library,

pygame can work on all platforms and operating systems. Tkinter is a standard graphics library that allows the game to be played with continuous updates. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

II. RELATED WORK

In the following literature review we went through multiple research papers. Going through the paper we came to know many concepts related to the pac man game. We examine several key approaches for describing the composition of games and comparison of different search algorithms for the . Also how the game can be implemented in 3d version unlike the B&W retro game on television. Finally we also mentioned other notable research papers related to the project.

In this paper [1], "Design of Pac-Man" implemented by Ke, Eric and Winston developed this project for reconstructing the Pac-Man game. In this, the project is split into software and hardware programming, and they have reduced hardware complexity with little effort on the software side. In this project, Game logic like collision detection, scoring and enemy artificial intelligence (AI) are included and are implemented using the MicroBlaze. It implements a graphics engine with sprites and paddle control over the MicroBlaze UART IP Core and Hardware configuration is described in Very Large Scale Integration HDL (VHDL)

In this paper [2], "The Pac-Man project" by Raymond Chen and John Martin L was to replicate the great arcade classic Pac-Man on television in black and white in a 3-d Version which can be used on any television inputs. This project used an Atmel AT90S8515 microcontroller and a simple video DAC to produce our Pac-Man game on any television with composite inputs.

In this paper [3], "Pacman Game Using Java", they used a recursive Backtracking algorithm used in game theory. This project uses Java to discuss the popular Pacman game. With the maze, a Pac-man, and Pac-dots for the Pac-man to eat, they created a simple Pacman implementation. They made a total of eight Pac-man images for this, one of which changes direction and has an alternating open/closed mouth. The Pac-man consumes the Pac-dots by moving around in legal positions at random. Several ghosts were added. This second increment was accomplished by checking to see if the game has ended (ghost catches Pacman or Pacman consumes all of the dots). Although the graphical representation is the most visible aspect of the game, it is implemented utilizing multi-threading and synchronized methods and algorithms from Data Structures.

In this paper [4] "Artificial Intelligence based Pacman Game", The goal was to create a Pac-man game using AI. Pac-Man is a difficult video game that can be helpful in AI (Artificial Intelligence) research. They have created several AI algorithms for the pacman game because it aids in the study of AI through the use of visualizations, which allows them to understand AI more effectively. The basic purpose is to create an intelligent Pacman agent that determines the best paths through the maze to achieve a specific goal, such as finding a specific food spot or avoiding ghosts. They've used AI search algorithms including Depth first search, Breadth first search, A*search, and Uniform cost search to accomplish this. Multi-agents such as Reflex agent, Minimax agent, and Alpha-beta agent have also been implemented. They can make Pacman react to its surroundings and escape from ghosts using these multiagent methods.

In this paper [5] this study proposes a model-based approach for computing real-time optimal choice strategies. Ms. Pac-Man is a great benchmark problem for a chase-evasion game with several, active adversaries that adjust their pursuit policies in response to Ms. Pac-condition Man's and decisions. The agent must chase various fixed and movable targets in an obstacle-filled environment in addition to avoiding the enemies. This work proposes a novel method for deriving a decision-tree representation of all feasible strategies from the maze geometry and the adversaries' or ghosts' dynamic equations. Extensive numerical simulations are used to validate the suggested models of ghost dynamics and decisions. The decision tree is updated and utilized to find optimal strategies throughout the game.

In this paper [6] "Learning to Play Pac-Man: An Evolutionary, Rule-based Approach", this paper describes the first method of development, a man-made agent who plays the simplified version of Pac-Man. The agent is specified as a simple limited position machine and set of rules, with limits that control the probability of movement by the agent is given maze limits instantly time. Contrary to previous methods, the agent stands for evolving Pac-Man gaming strategy, instead of a pre-arranged maze solution.The agent "reads" flexibly by adjusting agent parameters via population-based incremental learning (PBIL). Test results are available that

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

provide insight into some of the complexities of the game, and highlight the limitations as well the difficulty of agent representation.

In this paper [7], "Pacman Project", The Pac-Man projects apply an array of AI techniques to playing Pac-Man. The project mainly focuses on AI concepts like informed state-space search, probabilistic inference, and reinforcement learning. The projects enable you to see the outcomes of the approaches you use. They also include code examples and straightforward instructions without requiring you to dig through excessive scaffolding. Finally, Pac-Man presents a difficult issue setting that necessitates imaginative solutions; real-world AI challenges are difficult, and Pac-Man is no exception.

III. SOFTWARE ANALYSIS

Software model used in the project is the agile model. Agile refers to something that is quick or adaptable. A software development approach based on iterative development is referred to as an "agile process model. Agile approaches divide projects into smaller iterations or sections and avoid long-term planning. The scope and requirements of the project are defined at the start of the development phase. The number of iterations, duration, and scope of each iteration are all clearly determined ahead of time.As we used an agile model the project was divided into smaller tasks which were like building stones for the project .Completing each task made little progress in the project . It was easy to do projects according to task as each concept was cleared in the particular tasks.

Fig 3.1: Phases of Agile model

1. Project Initiation: The beginning of the agile software development life cycle. This initial stage, also known as the inception or envisage phase, focuses on discussing the project vision. This is a high-level discussion of feasibility that does not get into specifics. This step entails identifying team members as well as determining the amount of time and work resources required to accomplish the project. Taking inventory of resources is critical for establishing the economic feasibility of a project before it can be approved.

2. Planning: The Agile lifecycle takes shape for the team during this exploratory phase. We talk about how growing the backlog at the story level will help us do this.. The type of user, what they want from the product, and why should all be included in the story. It's important to think about the business possibility in a larger framework. This will have an impact on the project's functional and financial viability. With an initial release strategy, you should estimate the risks and set milestones. Only when your backlog is complete and you've prioritized the items based on business value and dependency is planning complete.

3. Development: The actual work begins after the requirements have been developed based on feedback from the product owners and stakeholders. In incremental phases, sprints, or iterations, agile product development produces high-quality working products. Developers begin work on the first version of the product with the goal of completing the sprint with a working, usable product. This is not the final version and will go through several modifications, therefore it should only offer the bare minimum of features. This capability can be enhanced in future Agile lifecycle iterations. Teams can succeed in these sprints by ensuring team and stakeholder collaboration. Following code rules and style requirements to maintain quality. Following the priorities stated by stakeholders with complete control

4. Release: Your product has now been released and is in use by end users. It's critical to keep an eye on these early phases for faults or defects that were missed during testing. Between the production and support teams, there should be a handover with appropriate training. Depending on the type of product you're producing,

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

these final processes and handovers may differ. When the product is ready to be retired, the production phase usually finishes.

IV. DESIGN AND IMPLEMENTATION

Objective of the project is to create a maze inside which game is played The goal of the game is to eat all of the dots placed in the maze while avoiding four colored ghosts or spirits that pursue him. Create a pacman which is controlled by player using arrow keys When Pac-Man eats all of the dots, the player wins the game .If Pac-Man makes contact with a ghost, he will lose a life the game ends .To improve the overall user experience

In the era of AAA and high speed games with amazing high graphics we wanted to create a legendary arcade game pac man which is famous from many decades but the new generation have started to forget the game. In addition to classic retro games we have tried to add better graphics and made it a bit easy to play for the new generation. There are many different versions of pac man game but we created a Classic version of pac man Our game version has a pac man that is the player itself and 4 ghosts which are named as Blinky, Pinky, Inky and Clyde collectively known as Ghost Gang. Game is kept quite easy to play so that even younger children can enjoy playing it.

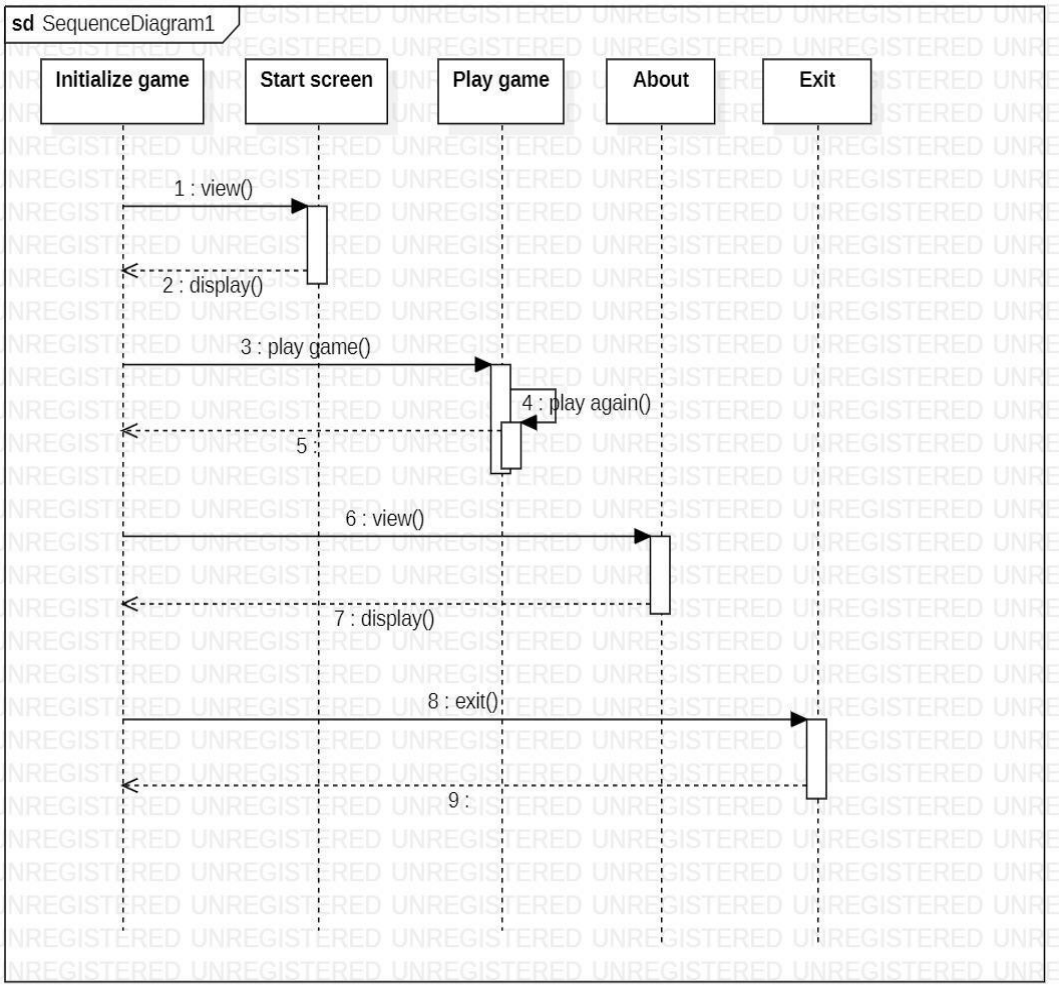

Fig 4.1: Sequence diagram

Sequence diagrams depict how items interact with one another and in what sequence they do so. A sequence diagram depicts item interactions in chronological order. It illustrates the scenario's objects and classes, as well as the sequence of messages sent between them in order to carry out the scenario's functionality. The lifeline is a line that depicts how long an object has been present in a process. After the game is initialized it shows the start screen and the main menu is displayed. In which a player can choose whether to play a game or to check about the game or exit the game. After clicking on the play game button the game is initialized and the pac man and ghost are initialized at the start positions.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

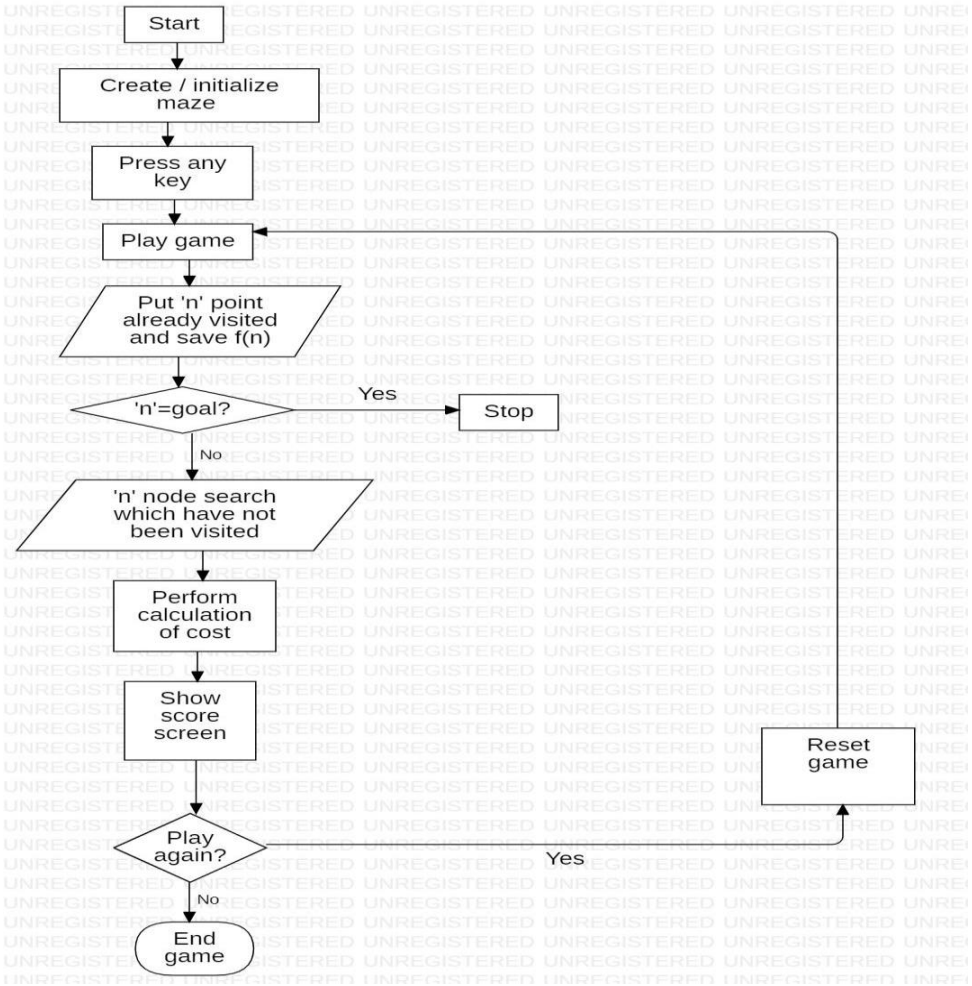

Fig 4.2: Flow Chart Of Game

The two search algorithms which we studied are Uniform Cost search algorithm and A* Algorithms.

A* Search: An educated search algorithm with prior knowledge of the problem is known as an A* search. A* is a best-first-search algorithm using the formula f(n) = g(n) + h(n), where f(n) = g(n) + h(n).

g(n) = total cost of edges from start to n

h(n) = lowest-cost path from n to goal estimate

f(n) = current distance + estimated remaining distance

If h(n) underestimates the cost of any solution that can be obtained from n, it is said to be acceptable. If C*(n) is the cheapest solution path from n to a destination node, and h is permissible, then

h(n)<=C*(n)

If h(n) is admissible, we can show that the search will find an optimal solution.

[8].

OPEN = nodes on frontier.

CLOSED = expanded nodes.

OPEN = {<s, nil>}

while OPEN is not empty

remove from OPEN the node <n,p> with minimum f(n)

place <n,p> on CLOSED

if n is a goal node,

return success (path p)

for each edge connection n & m with cost c

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

if <m, q> is on CLOSED and {p|e} is cheaper than q

then remove n from CLOSED,

put <m,{p|e}> on OPEN

else if <m,q> is on OPEN and {p|e} is cheaper than q

then replace q with {p|e}

else if m is not on OPEN

then put <m,{p|e}> on OPEN

return failure.

Uniform Cost Search: In uniform cost search we enqueue nodes by path cost. Let g(n) = cost of the path from the start node to the current node n. The algorithm sorts nodes by increasing the value of g, and expands the lowest cost node of the fringe.

Properties of Uniform Cost Search

- Complete
- Optimal/Admissible
- Exponential time and space complexity, $O(b^d)$

The UCS algorithm uses the value of g(n) to select the order of node expansion. We will now introduce informed search or heuristic search that uses problem specific heuristic information. The heuristic will be used to select the order of node expansion.UCS takes more time and cost to expand the nodes and find the cost

While creating pacman project we created following modules:

1. main.py
2. enemy.py
3. player.py
4. game.py

main.py

This is the main file of the project.

In the following module we import all the pygame modules.

Set the height and width of the game window.

Create main menu of the game including

- Start game
- About
- Exit

enemies.py

In the following module we create the ghosts of the game we define shape and color of the slime i.e ghost Also we create a maze inside which the slime moves to catch the pac man. Define the moments of the slime inside the maze.

player.py

In this module we create a pac man for that we import the image of pacman. Load a image which will be the animation of the moment of pacman Flip the image of pacman if it changes its direction from top to bottom and vice versa. Rotate the image of pacman if it changes its direction from left to right. Load explosive images which will appear when pac man dies.

game.py

Player and Enemy modules are imported in game.py along with a message box from tinker. A variable to display score is created which goes in incrementing when pacman eats the pills in the maze. initial position of the pac man and ghosts are defined a group of dots that are pills are created in a whole maze. sound effects of the game are loaded in the game using pygame. mixer. arrow keys are assigned to the pacman for its up, down, left, right moments in the maze.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

V. RESULT ANALYSIS

Game is implemented in the python programming language. We choose python because it has many built in libraries. Python's distinct syntactic structure distinguishes it from other programming languages for game development. Python code is easier to grasp than Java or C code, almost any developer will agree. Python is a popular choice among game developers due to its flat learning curve. Python is object-oriented, includes built-in high-level data structures, and enables dynamic type and dynamic binding. Python does allow game creation, albeit it isn't as popular as C++ with DirectX and OpenGL. We used the Pygame library of python. PyGame is a game-making library that is both developer-friendly and simple to use. Python is a simple language to learn, and creating games in Python isn't difficult either. For the moments of the ghost in the maze we studied the comparison between two search algorithms and decided which among the two is best for chasing the pac man in the maze.

Check performance of search algorithms

Table 5.1: Comparison between USC and A*

Algorithm	Maze	Node expansion	Cost	Time
UCS	bigMaze	620	210	7.3s
A*	bigMaze	549	210	0.6s
UCS	mediumMaze	269	68	1.6s
A*	mediumMaze	221	68	0.1s

The only difference between A * and UCS searches is UCS looks at the cost of accessing the location where A * Search looks at the total cost of access to the site and the cost of access to the goal node from that node (g + h) instead of g. Here, the heuristic function used is the Manhattan range heuristic. This heuristic is used to determine which node or state is closest to the target state.

VI. CONCLUSION

The primary goal of the project was to create a Retro Classic pacman game for the purpose of entertainment and which can be played by all age groups. Pac man which finds its origin back in 1980 created in Japan is one of the most famous arcade games .The project which we undertook has helped us to gain a better perspective on various aspects related to our course of study as well as particular knowledge of particular cmd-based applications.

According to this project, we understand some of the key algorithms, heuristic search, and specific game theory. Based on this information, we use python to design two different agents. One can be a pacman and eat the food of rivals, and one can always be a defensive ghost, preventing the pacman from eating our food. Basically we use two different algorithms for this project, A* search and UCF. Each algorithm has its advantages and disadvantages. Finally we selected A* because of its less time complexity and its more efficient in finding the goal node.

The player who succeeds in eating more pills by avoiding ghosts scores many points. Game can be won when pacman eats up all the food pills without getting killed by 4 ghosts. The children who play the pacman game can improve their concentration and have a strategy in solving the problem.

In conclusion, Games can be described as a very big thing. The game is able to make his children always play creatively and do not abuse the game. The children who play the pacman game can improve their concentration and have a strategy in solving the problem. It is kept easy and interesting so it won't be boring for younger children. We have made use of vibrant colors so the screen seems interesting which was the goal of the original project too.

ACKNOWLEDGMENT

We would like to express our sincere gratitude towards our guide Dr. Seema Biday, Prof. Pramila Mate and Prof. Randeep Kahlon for their help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stint thoroughly pleasant and enriching. It was great learning and an honor being their student. We are deeply thankful to Dr. Archana Mire (HOD of Computer Department) and entire team in the Computer Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work. We take the privilege to express our sincere thanks to Dr. L. K. Ragha our Principal for providing encouragement and much support throughout our work.

VII. REFERENCES

[1] Ke, Eric and Winston, "Design of Pac-Man," IEEE, 2014.
[2] Raymond Chen and John Martin L, "The Pac- Man Project," IRJET Volume:04 Issue:2015.
[3] Uthkarsh Ch., "Pac-man Game using Java," Kakatiya Institute of Technology & Science, Warangal(A.P.) 2018.
[4] Himakar Sai Chowdary Maddipati, Aravind Kundurthi, Pothula Mahima Raaj, Kapudasi Srilatha, Ravi Kishen Surapaneni, "Artificial Intelligence based Pacman Game," International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075 (Online), Volume-9 Issue-9, July 2020.
[5] Greg Foderaro, Ashleigh Swingler, Silvia Ferrari, "A Model-based Approach to Optimizing Ms. Pac-Man Game Strategies in Real Time," School of Information Technology and Electrical Engineering University of Queensland 4072 Australia 2016.
[6] Marcus Gallagher, Amanda Ryan, "Learning to Play Pac-Man: An Evolutionary, Rule-based Approach," IEEE Transactions on Computational Intelligence and AI in Games DRAFT - JANUARY 27, 2016.
[7] John DeNero, Dan Klein, "Pacman Project" UC Berkeley CS188 Intro to AI, 2015.
[8] "Problem Solving using Search- Single agent search," Version 2 CSE IIT Kharagpur, 2018.