# Project Proposal

## Team Name

Gentian Monsters (GM)

## Team Composition

|  |  |  |
| --- | --- | --- |
| Student Names | Student Numbers | NetIDs |
| Grant Keefe | 20266090 | 20gak5 |
| Ryan Berry | 20277012 | 20rpb1 |
| Maxwell Dizy | 20271347 | 20mld4 |
| Gabriel David | 20261837 | 20gad4 |

## Title of Project

“Maze Runner: the impossible level”

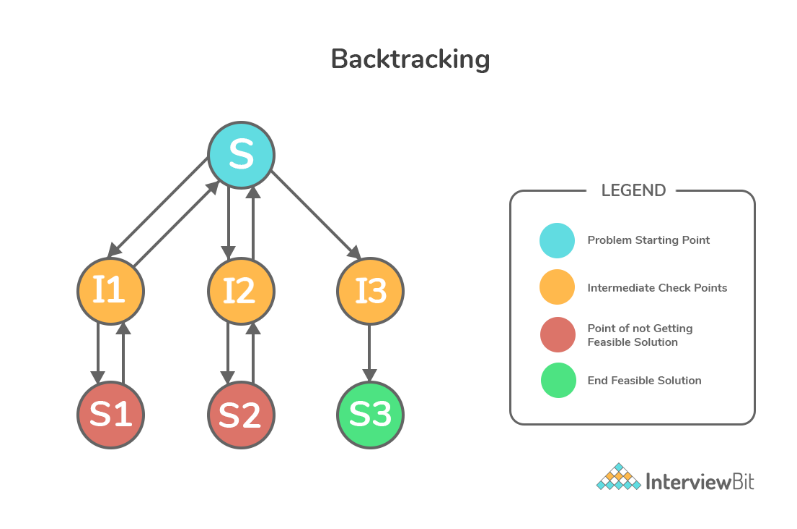
## Brief Description of Project

Maze Runner: the impossible level, is the video game stemming from the Maze Runner movie series. This game is played on a checkered board scattered with obstacles such as holes, trapdoors, or walls. The objective of the game is for the player to reach the indicated “point b” from the starting position “point a”. There are pre-specified traversal rules such as a knight in chess that will change through each level. The player will be allowed to move first, then the computer, and so on. The computer, however, will take the optimal path to “point b” resulting in the player losing the game if they make one wrong move. Each level, the board will get bigger, increasing the difficulty for both the player and computer to generate the optimal path from “point a” to “point b”. The game will continue until the computer beats the player to “point b”, you can no longer see the squares of the board due to laptop screen sizes, or the player dies of old age.

## Preliminary Design for Project

First the ‘rules’ of how a runner is allowed to travel on the board will be generated and displayed to the player. They will then make the first move towards “point b”, followed by the computer. The two runners will take turns moving one move at a time until one runner reaches “point b”. If the player wins, the board will increase in size and the next level will commence with a different travel ‘rules’. If the computer wins the player will explode and the program will end. A breadth first search will be used to backtrack through the graph and find the best combination of moves to get to a desired space. A diagram outlining how backtracking works is included below.

Diagram from: <https://www.interviewbit.com/courses/programming/topics/backtracking/>



## Course Topics Covered

-Queue

-Trees

-Sorting Efficiency

-Asymptomatic Notation

-Graphs

## Project Schedule

March 4th – Git Repository In VS-Code

March 6th – basic game GUI and can accept user input

March 8th – Working graph that can represent the possible moves of the runner

March 12th – runner can move to spots but not necessarily fastest way

March 28th – runner moves to the desired square in the least amount of moves

April 1st – Correct Big O, Omega, and Theta Notations representing the complexity of the algorithm

April 3rd – Final testing and debugging completed

Project Demonstration and 1st Code Submission – April 4th

April 7th – cleaning up / refining code making it as streamlined as possible, debugging and finishing commenting out code

Final Project Report and 2nd Code Submission – April 8th