Defuse the Bomb

A CSC 102 Project

Team: Maze Runners (Group 2)

BOMB DEFUSAL MANUAL

Version 1

Verification Code: <enter your code>

**The Game**

This project is based on the game **Keep Talking and Nobody Explodes**[[1]](#footnote-2), a cooperative bomb defusing party game. As the game designers put it, “You’re alone in a room with a bomb. Your friends, the 'Experts', have the manual needed to defuse it. But there’s a catch: the Experts can’t see the bomb, so everyone will need to talk it out – fast! Put your puzzle-solving and communication skills to the test as you and your friends race to defuse bombs quickly before time runs out!”

Their version is a software game. Our version takes the idea and realizes it as a physical device with buttons, switches, and more! Although our version can be played just like theirs, players can interact with both the bomb and this document at the same time (i.e., players can both defuse the bomb and serve as the “Experts”, using this document to help disarm the phases).

The backend of our version of the game is a Raspberry Pi[[2]](#footnote-3) computer that combines a typical computer with the ability to interact with the outside world through sensors. The underlying software is written in Python[[3]](#footnote-4) and is the result of a final group-based project in CSC 102 (The Science of Computing II) in the Computer Science Program at the University of Tampa.

**Defusing Bombs**

The bomb will “explode” when its countdown reaches 0:00 or when too many strikes have occurred. You defuse the bomb by disarming all its “phases” before the countdown expires.

**Game Overview**

Inspired by the mechanics of Keep Talking and Nobody Explodes, this version integrates bomb defusal with a maze exploration challenge. Players must navigate the maze, solve puzzles, and defuse the bomb before time runs out

**GUI Customization**

* Interactive layout featuring a maze map.
* Speaker integration for sound effects (alerts, countdowns, success prompts).
* Image-based elements for toggles, buttons, and keypad interactions.

**Game Structure**

* Players begin in the maze.
* Must solve challenges using toggles, wires, buttons, and keypad.
* Failure results in a strike, reducing time until detonation.
* Bonus challenges extend time.

**Phases**

The challenge consists of four distinct phases, each requiring specific actions to successfully navigate the maze and ultimately defuse the bomb. Phases can be completed in any order, but all must be disarmed before the timer runs out. Each phase presents a unique puzzle—whether it’s toggles for movement, keypad riddles, wire cutting, or button interactions. Once a phase is completed, it locks in place, ensuring no accidental resets.

**Strikes**

Mistakes in solving puzzles or inputting incorrect sequences result in **strikes**. Accumulate too many strikes, and the bomb will **detonate**, ending the game. Additionally, receiving a certain number of strikes may accelerate the timer, forcing players to act quickly and strategize carefully. Every choice matters, so accuracy is crucial!

**Information**

Each time the game begins, a **randomized version** of the bomb is generated, ensuring no two sessions are identical. With **6,720 unique bomb configurations** and over **1 million possible variations**, players must carefully analyze each instance. Some phases require specific details from the maze’s boot-up text—such as numeric codes, color patterns, or direction-based hints—to proceed. Observing and recalling this information will be the key to victory!

Regarding the Toggles

**Toggles (Maze Movement)**

* Toggle switches control movement (North, East, South, West).
* Must follow a pattern based on hints before moving.
* Incorrect toggles result in time penalties.

Toggles play a critical role in navigating the maze, acting as directional controls that determine the player’s movement. Each toggle switch corresponds to a compass direction—North, East, South, and West—allowing players to progress through different points in the maze. However, movement isn’t entirely straightforward; instead of simply flipping a switch and moving in the corresponding direction, players must decipher clues or follow a specific pattern to determine the correct sequence. For example, they may encounter riddles or hints that subtly guide them toward the proper toggle, encouraging problem-solving before they can advance. Incorrect toggles result in time penalties, making precision and logical thinking essential for success. Additionally, toggles could be paired with environmental changes—such as walls shifting or obstacles appearing—to further challenge players to anticipate their next move wisely. Some stages may even feature a hint system that provides subtle guidance to prevent unnecessary difficulty while maintaining the puzzle-solving nature of the mechanic. By integrating toggles into the movement system, the game encourages players to approach navigation with strategy and awareness, reinforcing the immersive nature of the maze-runner theme.

Regarding the Button

**Buttons (Timed Challenges & Bomb Defusal)**

* Button flashing sequences prompt bomb interaction.
* Players must decipher how and when to press based on color-coded mechanics:
  + Green: Simple numeric entry.
  + Red: Advanced converting from binary to decimal.

Buttons introduce a high-stakes, reaction-based element to the game, requiring players to carefully follow instructions to press and release them at the right moments. Unlike toggles, which are used for movement, buttons serve as interactive triggers that unlock doors, initiate challenges, or even determine the fate of the bomb. Each button is surrounded by a lighted ring that can flash in different colors—red, green, or blue—each with distinct interaction rules. Players must press the button when prompted but must release it according to the correct sequence, depending on its color and related conditions, such as time constraints or countdown digits. In tougher challenges, flashing buttons may appear rapidly, increasing difficulty as players must remain focused under pressure. Incorrect timing or premature release results in strikes, reducing available time and bringing the game closer to an explosion. Additionally, buttons could be incorporated into maze interactions, such as activating hidden pathways or requiring a sequence of presses to complete a puzzle. Some stages may present players with a choice—pressing a red button could trigger a Caesar shift challenge, while pressing a green button might allow for an easier code entry task. The unpredictability of button mechanics forces players to stay sharp, analyze their surroundings, and make quick yet calculated decisions to progress and ultimately defuse the bomb.

Regarding the Keypad

**Keypad (Code Entry & Riddle Challenges)**

* Enter numeric codes to open doors.
* Solve riddles related to maze themes (e.g., Boolean logic puzzles).

Keypads serve as intellectual obstacles in the game, challenging players to input codes, decrypt words, and solve riddles to unlock doors or progress through different phases. Unlike buttons and toggles, which rely on timing and movement, keypad interactions require logical reasoning and pattern recognition. Some challenges involve entering a direct code from a clue or converting sequences like binary numbers into decimal form, forcing players to engage in quick mental calculations. Other puzzles introduce substitution ciphers, where a keyword must be decrypted using a shifting pattern before cross-referencing a table to determine the correct passphrase. The pressure increases when mistakes result in strikes, reducing time and heightening the urgency to solve each sequence correctly. In more advanced stages, keypads might feature multi-layered puzzles—requiring players to answer riddles before gaining access to numeric codes—or could dictate directional movement based on correct inputs. The complexity and variability of keypad challenges ensure that players remain engaged, combining cryptographic skills with fast decision-making to overcome obstacles and ultimately defuse the bomb

Regarding the Wires

**Wires (Barriers & Power Gates)**

* Power barriers randomly placed throughout the maze.
* Must cut wires in the correct sequence to proceed.
* Hints provided to prevent excessive difficulty.

Wires act as critical barriers throughout the game, challenging players to carefully determine the correct sequence for "cutting" or leaving them intact. Unlike toggles and buttons, which focus on movement and timed interactions, wires introduce a strategic layer where logic and deduction play a key role. Some phases require players to analyze clues based on the bomb’s serial number or button colors, ensuring they make precise choices that won’t trigger an explosion. In the maze, power barriers controlled by wires appear at unexpected locations, forcing players to engage with puzzles before continuing their escape. Certain challenges may require recognizing letter sequences or patterns to select the right wires, while others rely on process-of-elimination mechanics to ensure the maze remains dynamic and engaging. Strikes are given if an incorrect wire is cut, reducing available time and increasing pressure to solve the puzzle efficiently. Additionally, some stages provide subtle hints to assist players without making solutions too obvious. By integrating wire mechanics into the maze structure, the game keeps players on edge, reinforcing the high-stakes atmosphere while maintaining an immersive puzzle-solving experience.

1. <https://keeptalkinggame.com/> [↑](#footnote-ref-2)
2. <https://www.raspberrypi.com/> [↑](#footnote-ref-3)
3. <https://www.python.org/> [↑](#footnote-ref-4)