|  |  |
| --- | --- |
| **Team Name** | **Team Members** |
|  |  |

|  |  |
| --- | --- |
| **Step 1:**  **Identify and Define** | **Write your responses in these blocks.** |
| Write a real-world, global perspective challenge statement for this project. |  |
| What are the **requirements?**  *(What do you want to happen?)* |  |
| What are the **design** **constraints**?  *(Describe* limitations *on programming your robot.)* |  |

|  |  |
| --- | --- |
| **Step 2:**  **Research** | **Write your responses in these blocks.** |
| Thoroughly **describe**, in paragraph form, **all** the information needed to design a successful program. Don’t simply list the information. **Explain** why the information is useful in designing a successful robot program. Include all information from ELEGOO tutorials and testing that were necessary to make your robot successful. Make updates as necessary. |  |

|  |  |
| --- | --- |
| **Step 3:**  **Generate Ideas** | **Write your responses in these blocks.** |
| Using graph paper, **sketch** and **describe** three different paths that the robot can take to the rescue point. Use **graph paper** for your sketches. |  |

|  |  |
| --- | --- |
| **Step: 4**  **Selection Process** | **Write your responses in these blocks.** |
| Share and discuss the ideas among team members.  **Decision Matrix**  Describe four to six criteria used in your solution selection process.  Fill in the decision matrix with criteria and proper design idea names. Rate each design appropriately. The criteria should be measurable **before** you test your program on the field.  Come to a **consensus** on what design your team is going to move forward with. | Provide a description of each criteria:  Provide a description of the rating scheme for each criteria:  **Decision Matrix**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **Criteria 1** | **Criteria 2** | **Criteria 3** | **Criteria 4** | **Total** | | **Path 1** |  |  |  |  |  | | **Path 2** |  |  |  |  |  | | **Path 3** |  |  |  |  |  |   Describe the features (path, movements, programming functions) you plan to incorporate in your final design solution. |

|  |  |
| --- | --- |
| **Step 5:**  **Implement Solution** | **Write your responses in these blocks.** |
| **Part A**  Create a graph of your selected path using Excel overlaid on a photo of the obstacle course  Provide a table, including the starting point and end point, of all the points that your robot follows along its path. | *Final Path Example*   |  |  |  | | --- | --- | --- | |  | **X (ft)** | **Y (ft)** | | Origin | 1 | 0.8 | |  | 4.5 | 0.5 | |  | 5.8 | 3 | |  | 5.8 | 4.7 | |  | 7 | 5 | | End Pt | 7.5 | 7.5 | |  |  |  | |
| **Forward Speed**  Write a detailed procedure on how you determined the forward speed of your robot.  Include a Table of data collected. Must include at least five data points.  Include a Distance vs Time graph of your data. Determine the slope of the best-fit line. |  |
| **Turning Speed**  Write a detailed procedure on how you determined the turning speed of your robot.  Include a Table of data collected. Must include at least five data points.  Include a Distance vs Time graph of your data. Determine the slope of the best-fit line. |  |
| Show a sample calculation for the time delay for driving forward and the time delay for turning.  Include a complete table of all robot path calculations with times. Include the Total Time for Path.  Write out the pseudocode of your robot path with specific time values. | Forward Drive Time Sample Calculation:  Turning Time Sample Calculation:   |  |  | | --- | --- | |  |  | | **Forward Distance (ft)** | **Forward Time (s)** | | 3.513 | 2.927 | | 2.818 | 2.348 | | 1.700 | 1.417 | | 1.237 | 1.031 | | 2.550 | 2.125 | |  |  | |  |  | | **Relative Angle (degrees)** |  | | -4.90 |  | | 62.53 |  | | 90.00 |  | | 14.04 |  | | 78.69 |  | |  |  | |  |  | | **Turn Angle (degrees)** | **Turn Time (s)** | | -67.42 | 1.568 | | -27.47 | 0.639 | | 75.96 | 1.767 | | -64.65 | 1.504 | |  |  | |  |  | | **Total Time for Path (s)** | 15.325 | |  |  |   *Example table of robot path calculations.*  Pseudocode of Time-Based Programming: |
| **Part B**  Document problems and solutions about the programming process. | Problems/solutions of programming the robot: |

|  |  |
| --- | --- |
| **Step 6:**  **Evaluate Solution** | **Write your responses in these blocks.** |
| Video record your TurboPi robot completing the course. Send the video to your instructor.  While watching the video, and using a stopwatch, determine the actual time it takes to travel through the path without hitting any obstacles.  Describe ways you can possibly improve your robots speed to the rescue point.  **Include an entire copy of the final program in Attachments section. Be sure to include proper comments throughout.** | Time and date of sending video to instructor:  Actual Time through Path:  Description of improvements: |

|  |  |
| --- | --- |
| **Step 7:**  **Refine (#1)** | **Write your responses in these blocks.** |
| **Refinement #1:**  What changes did you make to improve your design?    Make these changes when you are ready.   **Be aware of your time.**  **Include the final refined code as an attachment.** | **Refinement #1:**   * What did you do and WHY?   Results of refinements (include Vidigami URL): |

**Attachments**

**Final Solution Python Code with proper comments throughout:**