|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Your task is creation of the simulation of a robotics competition.     |  | | --- | | The competition consists of three phases.   1. Speed. The robots will race to reach the finish line. 2. Strength. The robots will attempt to lift several heavy objects. 3. Combat effectiveness. The robots will battle against 100 humans and then receive a score based on the number of victories. |     Design and implement the following:     1. Create a Robot struct with the following fields:    * Robot\_Number    * Robot\_Name    * Year\_Manufactured    * Top\_Speed    * Mass     Consider the following list of robots that could possibly be used in simulation of a robotics competition.    Table 1: List of robot information   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Robot**  **Number** | **Robot Name** | **Year**  **Manufactured** | **Top Speed**  **(m/s)** | **Mass**  (kg) | | 1 | AdamTheLink | 1999 | 15 | 110 | | 2 | BrainAT | 2010 | 32.5 | 18 | | 3 | BOBBY | 2013 | 29.5 | 7.5 | | 4 | MiMi | 2018 | 40.0 | 8.8 | |  | … | … |  |  |        1. Extend your previous program provided to simulate the robotics competition using the robot information stored in the Robot\_info array.     Your program should first create an array of structs, Robot\_info [ ], that stores robot information (using above defined struct, Robot). Your array should be able to store information for up to 50 robots. Initialize the Robot\_info array with appropriate robot information for at least 5 robots. The program then displays all available robot information (with a properly designed format) on the screen. |

|  |  |
| --- | --- |
| The program then selects a list of robots for a scenario of robotics competition:     1. Prompt the user to enter 2 variables: finish\_line\_distance (in m) and finish\_time (in sec.). The program then calculates a *Trial\_speed*, using the variable values entered (and formula from previous workshops). 2. The program searches robot information from the Robot\_info array to decide which robots are capable of being used for the scenario of robotics competition. That is, for every robots in the array, if its *Top\_Speed* is greater than or equal to the *Trial\_speed*, the corresponding robot is considered being capable of being used for the robotics competition, and its information are displayed on screen.     It is up to you to decide the format to display you search result. For example, it may contain below information, like     |  | | --- | | **Robots capable of being used for a robotics competition with Trial\_speed of <TS> (m/s):**  **---------------------------------------------------------------------------------------------------------------------------------**  **<robot 1 information>**  **<robot 2 information> … ...**  **<robot k information>**  **----------------------------------------------------------------------------------------------------------------------------------**  **A total of <*number*> robots are capable of being used for the robotics competition scenario** |   where **<TS>** is the Trial\_speed calculated above, and <number> is the *total number* of robots stored in the Robot\_info[ ] that could be used for the competition.  Note: Your program should display an error message if something goes wrong, for example, to deal with the following case/s:   If any of these variables finish\_line\_distance and finish\_time is entered 0 or less, etc.  Imp\* - flowchart design of functions (how they operate)  Further extend your program to:   * Add a menu that not only do the above task but also extra functions, such as (a) allowing users create (or enter information for) more robots. Use an integer to keep track of how many robots have been created so far.   Be careful – Don’t allow the user create more than 50;  (b) searching robot information by Robot Name, or by Robot Number, etc.   * Create a second array of robots to hold the search result for Step (ii), i.e., to hold information of robots capable of being used of the competition. * Explain and extend to accommodate more error messages, etc. |