Assignment 4   [30 points max]

Compulsory   [5 points max]

1. Create a program (in any programming language) for robot path planning in an In- **[5 points]**
   dustry 4.0 factory of a rectangular shape 80x24 m with several obstacles that the
   robots cannot pass through. Please do as follows. The
   program starts with the construction of a fine grid of
   80x24 interconnected points in E2, where the coordi-
   nates of the upper left corner are [0,0], the Euclidean
   distance between two adjacent points (in both horizon-
   tal and vertical direction) is 1 meter, and each point in
   this grid is interconnected with its eight closest points.
   The result is an undirected graph representing all the
   paths the robot can follow (with an accuracy of about 1
   m), as illustrated in the figure.
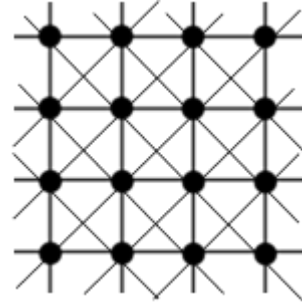
   For demonstration purposes, suppose to have just one obstacle, which is the circle
   with the centre at [5, 10] and a radius of 7 m, and four points of interest A, B, C, D
   with coordinates [0, 0], [0, 23], [79, 5], and [20, 15], respectively. The program re-
   moves all the points lying inside the obstacle from the graph (or makes these
   points inaccessible by other means), and after that, it finds the shortest (optimal)
   path between each pair of points of interest. **Note**: The program should be im-
   plemented generally. Finally, the program displays the result clearly on the console
   so that the letters A, B, C, and D will indicate the position of the points of interest,
   dots (.) the nodes covered by the obstacle, and asterisks (*) the nodes through which
   the optimal paths lead. The distances A-B, A-C, A-D, B-C, B-C, B-D, and C-D in meters
   to 1 decimal place will be listed on the last line. Therefore, the graphic output will be
   visible on a standard console (80x25 characters) without scrolling.

   Hint: draw a situational picture

Elective   [25 points max]

2. Try implementing the best robotic tank in the world, or at least one that will humili- **[15 points]**
   ate your ZEP-E opponents. Create your bot for the RoboCode Tank Royale applica-
   tion (version 0.30.1) − see https://github.com/robocode-dev/tank-royale/releases,
   which can defeat at least some out of 18 Sample bots (distributed with the applica-
   tion) in the **classic** fight of 10 rounds in the 800x600 arena (default). **Submit the
   screenshot** of the application showing the results for 10 rounds together **with all
   codes**. If your rank is 1 − 3, you will get 10 points, 4 − 6, 8 points, 7 − 9, 5 points and
   10 − 12, 2 points.

   If you implement your bot using Java, naming it with the prefix **ZEP_2025_**, the bot
   will qualify for the fight against the sample robots and the robots of your ZEP-E col-
   leagues. Only the first 10 positions will be ranked so that the player earns (11 - his
   ranking) bonus points, i.e. the first earns 10 bonus points. The overall sum of the
   points will be trimmed to a maximum of 15 points.

3. BONUS   **[10 marks]**

Register on https://www.hackerrank.com/ and solve any of the following challeng-es:

- Breadth First Search: Shortest Reach
- Jack goes to Rapture
- Chief Hopper

Full marks will be given if the code is sufficiently efficient and easy to follow. The proof of the solution (e.g., in the form of a screenshot from your HackerRank pro-file) must be submitted (together with the code) as a part of this assignment by the deadline.