# WALKING OVER A LINE OF BLACK & WHITE SQUARES

This project is divided into 2 parts. The total of both parts accounts for 20% of your final grade.

## PART 1: PROBLEM CASE GENERAL DESCRIPTION

Consider the following problem: you are given a path of N white and black squares. The exact distribution of white and black squares and the length of the path vary with the instance of the problem you are given to solve, and they are both known for each instance.

You start on the left-most square and the goal is to move off passed the right end of the path (i.e. passed the last square with no concern of how far beyond it) in the least number of moves. You can always move 1 square right, i.e. whether you are on a white or black square you can always move to the next square on the right. If you are on white, you can move 2 squares right. If you are on a black square, you can move 4 squares right. Every move has the same cost.

You are required to design and develop a system to solve this problem. The project is divided into different stages, each with its own set of learning goals, resulting in a total of 4 partial submissions until its final completion. As part of this programming project you will have to study the material covered in class and your textbooks to be able to answer the questions below. You will address the questions in an orderly manner according to the instructions you will be given for each of the project's partial submissions.

### Questions to be answered throughout part 1 of the project:

1. Identify the problem environment's properties and justify them. Define a performance measure for the system to be developed.

2. Formulate the problem as a search problem. Then reconsider the performance measure you already defined and justify whether it needs to be changed or not.

3. Consider BreadthFS and DepthFS. Say which of the following factors are known and if so, give their values:
   - Branching factor "b",
   - Maximum depth "m",
   - Shortest depth "d".

   According to your answer and to your performance measure, JUSTIFY which of these two methods, or both, are complete and optimal.

4. Propose a non-trivial heuristic for the problem that is both admissible and consistent (for example, h(n) = 1, constant, is trivial). You may use the relaxed problem technique if you like. JUSTIFY that your heuristic meets the ADMISSIBILITY and CONSISTENCY properties.

5. Taking into account all your previous answers, determine what heuristic search method (whether classical search or local search) is the best choice. JUSTIFY your answer.

# PART 2: PROBLEM CASE GENERAL DESCRIPTION

Consider the following change to the problem described in part 1 of the project: Initially, and also after executing a move, the system to develop can determine its current position on the array of squares and look at what color it is but it can not see the color of the squares beyond that point. The length of the array is known from the very beginning and the goal of the problem remains the same.

You are required to extend the system you have developed for part 1 in such a way that it can handle the new environment setting. There will be one unique submission for this part of the project that includes programming and answering the questions below.

## Questions to be answered throughout part 2 of the project:

1. Explain how the problem environment properties have changed in this new setting with respect to part 1 of the project.
2. Explain whether the problem can still be solved by search. If so, explain what kind of search method is applicable and what kind of solution it would find. Exemplify your previous explanation by drawing the first two levels of the generated search tree