**Homework 4 – Question 1**  
Adi Album 316563956 & Tomer Epshtein 313200750

Exercise 4.1 (B):

We modified the PRM implementation such that it computes paths with high vertical clearance.

We first implemented a function *vertical\_clearance* which calculates the vertical clearance of a given configuration.

Our modified PRM works as follows:

It samples configurations randomly and creates the usual PRM graph.

For example:

A picture containing text, stationary, envelope

Description automatically generated

Next, for every node (=configuration) we calculate its vertical clearance.

So we have an undirected graph with a function .

Now, by the standard PRM algorithm we search for some path with ‘shortest distance’ between source and target (if one exists). In our modified PRM we don’t care for distance but somehow want to find a path with large vertical clearance. We have two problems we must address:

1. Vertical clearance is defined on vertices, not on edges
2. Dijkstra finds shortest path by edge weights, not largest.

We define a term of weight on edges, denoted by , and defined by

Where if or we replace with some positive small .

By this definition we have the following:

An edge has small weight if both its adjacent vertices have large vertical clearance.  
An edge has larger weight the small the vertical clearance of its neighbors is.

This way we will prefer paths from source to target such that all configuration point in the path have high clearance.

Notes on implementation:

Finding a configurations vertical clearance isn’t trivial. We implemented a simple search method. Starting at a given configuration we take small steps up and down along the -axis until a first obstacle is revealed. The smaller steps we take the better resolution our vertical clearance will be (with an obvious tradeoff against the algorithm’s runtime).