An average 25-year-old human can run a mile in 10 mins during a 5k. I took the values for 5k run as it would be pretty representative as the paths given are also substantially long. Converting that values to m/s would give around 3 m/s. This is the base speed.

All the speeds for different terrain are stored as a number between 0 and 1, and multiplied with the base speed to get the actual speed. This is so that we can easily update the base speed and all others will automatically follow.

I assumed that the paved path to be the fastest to traverse. So, I took that as 1. I took the “Impassible vegetation” and “out of bounds” to be 0 as we can’t traverse through them. An average human swims with at 2 miles an hour. About a third as the speed on land. So, took that as 0.33. Foot path should be easy to walk on, hence took a value of 0.9. Rest I have looked at the image and made an assumption from the images.

The heuristic I had considered depends on the distance. The cost function is returned by the function called time in my program. This calculates the distance between the present pixel and the next pixel using the Euclidean distance formula, taking the x, y and elevation into consideration. Similarly, straight line distance from the next pixel and destination pixel is calculated, which would be my heuristic. I would argue the heuristic is decent considering the fact that even in the real world you would just know the general direction as to where to go and walk towards it, deviating slightly with respect to the immediate surroundings. I add both up and divide by speed to get a cost function in time. This is done as some paths can be of shorter distance but take longer as they can’t be harder to traversed. Taking speed into consideration, would eliminate that issue.

If the elevation is increasing, I took the cube of the elevation difference, instead of the square of Euclidean distance. This is to punish the paths with high elevations. This increases the cost exponentially with increase in elevation, which is a realistic scenario.

For winter, we need to convert all the water to ice if is within 7 tiles of land. For this I am traversing to each and every pixel and checking if it a water tile adjacent to a land tile. From this tile I can traversing 7 pixels in every direction and converting it to ice if it was previously a water tile. As it would slower to walk on ice, but faster than to swim in general, I took assigned it a speed of 0.8.

For spring, all the land tiles within 15 tiles of water and within 1 meter of elevation are converted to mud. For this I am traversing to each and every pixel and checking if it a water tile adjacent to a land tile. From this I am traversing 15 pixels in every direction and checking the that tile is within 1 meter of elevation. If yes, am converting that tile to mud. Traveling in mud would a lot tougher than to walk on land. But faster than swimming. Hence, I took a speed of 0.6.