Writeup:

For the different speeds through different terrains. I think it is very easy to walk on open land, paved road and footpath, so I set the speed of walking on them to 10. And I think it is very difficult to walk on rough meadow, so I set the speed of walking on it to 2. And I think it is pretty easy to walk on easy movement forest, but not as easy as open land, so I set the speed of walking on it to 8. And I think it is a little difficult to walk through slow run forest, so I set the speed of walking on it to 5. And I think it is more difficult to walk through walk forest, so I set the speed of walking on it to 4. And I think it is impossible to walk through impassible vegetation, lake swamp, marsh or out of bound, so I set the speed of walking on them to 0.

For the whole route, I find the path between two points one by one. About A* algorithm, I find the best neighbor for beginning node and keep the remaining neighbors. When I search the neighbors for this neighbor, I first find the best in remaining list. Then compare its neighbors with it, if I can find the better, it is the new neighbor, or the best in remaining list is the new neighbor.

For calculating the heuristic value, I calculate the 3D distance divided by the maximum speed. And it worked in my program.

For different seasons, I changed the original map to seasonal map.

In the fall, I change the speed of walking on easy movement forest from 8 to 9 through adding a new speed named "easy_movement_forest_fall".

In the winter, I first search the pixels which is "lake_swamp_march" but its neighbors are not all "lake_swamp_march", these are the edges of "lake_swamp_march". Then I change their neighbors which are "lake_swamp_march" to "winter_lake", the speed from 0 to 10. This means the pixels inward are set to can be walked on. Repeat this progress 5 times.

In the spring, I first find the edge points. For each points I have a queue, the first point is in the queue. Then I add its neighbors which are not in the lake or has the height difference less than one meter to the queue. I change their color to "spring_lake" and remove them from the queue. Then I find their neighbors in the same way. Repeat that for 15 times.