## A\* component

A non-playable character (NPC) uses an A\* pathfinding algorithm in our game. A\* uses nodes to determine the route. Therefore a node network is created at the begin of the game. A node network is built out cube primitives. Each node is a trigger which determine whether the node is accessible.

The NPC chooses a random accessible node to walk to. Then the algorithm calculates the cost for each node. This is done by the taking the absolute distance and add an estimated distance value (the heuristic part) to this. After that A\* determines which path has the lowest cost by taking the adjacent node with the lowest cost for each node.

The hardest part was making the node network suitable for multiple NPCs. This was disregarded in the initial script. To solve this problem, the node keeps record of the costs for all NPCs in arrays. The code had to be rewritten to accept this arrays.

## NPC interaction

Our game needed a way to interact with NPCs. If clicked on a NPC with a mouse, an interface (new Unity UI) is activated. Then a script reads out an XML-file to determine the content of conversation. It is possible for NPC to ask a question where the player has to respond to. It depends on the answer of the player how the conversation continues.

The major problem was making the conversation dependable on choices of the player. Therefore recursion was needed and this was difficult to implement.

Another problem was using the conversation interface if there were multiple NPCs in the scene. This was tackled by giving each NPC its own interface element.