Algorithm 1 A* Search Algorithm (Graph)

```
function A*(start, goal)
    $closedset \leftarrow the\ empty\ set$
    $openset \leftarrow start$
    $came\_from \leftarrow$ the empty map

    $q\_score[start] \leftarrow 0$
    $f\_score[start] \leftarrow q\_score[start] + heuristic\_cost\_estimate(start, goal)$

    while $openset \neq \emptyset$ do
        $current \leftarrow$ the node in $openset$ having the lowest $f\_score[\ ]$value

        if current = goal then return reconstruct_path($came\_from, goal$)

        remove $current$ from $openset$
        add $current$ to $closedset$

        for all $neighbour \in neighbour\_nodes(current)$ do
            if $neighbour \in closedset$ then
                continue

            $neighbour\_g\_score \leftarrow g\_score[current] +$ dist_between($current, neighbour$)
            if $neighbour$ not in $openset \parallel neighbour\_g\_score < g\_score[neighbour]$ then
                $came\_from[neighbour] \leftarrow current$
                $g\_score[neighbour] \leftarrow neighbour\_g\_score$
                $f\_score[neighbour] \leftarrow g\_score[neighbour] +$ heuristic_cost_estimate($neighbour, goal$)

                if $neighbour \notin openset$ then
                    add $neighbour$ to $openset$
    return $failure$
```