Algorithm 2 Safety-Aware A*

```
OPEN ← PriorityQueue()
CLOSED ← List()
START_NODE.g ← 0
START_NODE.f ← heuristic(START, GOAL)
OPEN.add(START_NODE)
while OPEN is not empty do
    current ← OPEN.pop_lowest_f()
    if current equals GOAL then
        return path
    for each neighbor in get_neighbors(current) do
        safety_cost ← height_risk + crowding_risk + angle_risk
        movement_cost ← distance(current, neighbor)
        tentative_g ← current.g + movement_cost + safety_cost
        if tentative_g < neighbor.g then
            neighbor.parent ← current
            neighbor.g ← tentative_g
            neighbor.f ← neighbor.g + heuristic(neighbor, GOAL)
            OPEN.add(neighbor)
```