
Algorithm 2 : $expandLevel(nodes, gLim(), gSum(), H) \rightarrow solved$

```
open = n | n ∈ nodes ∧ expandableThisLevel(n)
while open ≠ {} do
  n ← Pop(n ∈ open ∧ gSum ≥ H(n, dir(n)) + g(n))
  status(n) ← 'closed'
  for all neighbour in expand(n, dir(n)) do
    child ← (neighbour, dir(n), g + 1, open)
    if neighbour already exist in nodes then
      node = get(neighbour, nodes)
      if dir(child) == dir(node) then
        continue
      else
        return(True)
    end if
  end if
  if expandableThisLevel(child) then
    open+ = child
  end if
end for
end while
return(False)
```
