物资调配规划模拟系统

算法说明

Algorithm 1 Initialization of Routing Cost Matrix

```
Input: $G = (V, E)$
Output: $C$
$C \leftarrow \infty$, $n \leftarrow |V|$
for each $edge$ in $E$ do
    $C[edge.\text{from}][edge.\text{to}] \leftarrow edge.\text{cost}$
end for
// update the cost matrix from neighbor vertices
for $k = 1$ to $n$ do
    for $u = 1$ to $n$ do
        for $n = 1$ to $n$ do
            for $v = 1$ to $n$ do
                $C[u][v] \leftarrow \min(C[u][v], C[u][n] + C[n][v] + 100)$
                // 100 is the cost of on-vertex operation
            end for
        end for
    end for
end for
return $C$
```

Algorithm 2 Route Planning

```
Input: $G = (V, E)$, $s$, $t$, $q$
Output: $P$, $c$
$P \leftarrow \emptyset$
while not reach destination do
    $sortedNeighbors \leftarrow \text{sort(neighbors of last node, by cost to destination)}$
    for each $neighbor$ in $sortedNeighbors$ do
        if $neighbor$ not tried and has enough capacity then
            $P.\text{push}(neighbor)$ // try this neighbor
            break
        end if
    end for
    if no neighbor can be tried then
        $P.\text{pop}()$ // backtrace
    end if
end while
for each $node$ in $P$ do
    $c \leftarrow \min(c, \text{remaing capacity of node})$
end for
```