

```

1  Program Traveling Salesman
2  !
3  ! Given:  distance look up table
4  !          list of cities
5  !          path between cities
6
7  ! Permute all but home city
8  CALL permute(2, numCities)
9
10 ! Report best path and distances
11
12 ! Inner subroutine
13 permute(first, last)
14
15     ! Base case
16     if first == last then
17         initialize distance from home city (1) to path(2)
18         loop i = 2 to last-1
19             add to current distance the distance from path(i) to path(i+1)
20         endloop
21         ! Get distance from last city back home
22         add to current distance the distance from path(last) to path(1)
23
24         increment number of permutations
25
26         if this distance < best distance so far then
27             update best distance so far
28             save path to best path so far
29         endif
30     else
31         ! Mix it up
32         loop i = first to last
33             swap path(first) with path(i)
34
35             ! Recursion reduction step
36             call permute(first+1, last)
37
38             swap path(first) with path(i)
39         endloop
40     endif
41 end subroutine permute
42
43 end program

```