function POP(*initial, goal, operators*) **returns** *plan*

```
plan ← MAKE-MINIMAL-PLAN(initial, goal)
loop do
    if SOLUTION?(plan) then return plan
    S_need, c ← SELECT-SUBGOAL(plan)
    CHOOSE-OPERATOR(plan, operators, S_need, c)
    RESOLVE-THREATS(plan)
end
```

---

function SELECT-SUBGOAL(*plan*) **returns** S_{need}, c

```
pick a plan step S_need from STEPS(plan)
    with a precondition c that has not been achieved
return S_need, c
```

---

procedure CHOOSE-OPERATOR(*plan, operators,* S_{need}, c)

```
choose a step S_add from operators or STEPS(plan) that has c as an effect
if there is no such step then fail
add the causal link S_add --c--> S_need to LINKS(plan)
add the ordering constraint S_add ≺ S_need to ORDERINGS(plan)
if S_add is a newly added step from operators then
    add S_add to STEPS(plan)
    add Start ≺ S_add ≺ Finish to ORDERINGS(plan)
```

---

procedure RESOLVE-THREATS(*plan*)

```
for each S_threat that threatens a link S_i --c--> S_j in LINKS(plan) do
    choose either
        Promotion: Add S_threat ≺ S_i to ORDERINGS(plan)
        Demotion: Add S_j ≺ S_threat to ORDERINGS(plan)
    if not CONSISTENT(plan) then fail
end
```