```
BLOCK entry {
  margin <- §0§[];
  items <- §SELECT COUNT(*)
           FROM   lineitem AS l
           WHERE  l.l_orderkey = {0}§[orderkey];
  GOTO inter0
}
```

```
BLOCK inter0 {
  p1 <- §{0}§[items];
  GOTO inter2
}
```

```
BLOCK inter2 {
  IF §{0} < 1§[p1]
  THEN GOTO truthy0
  ELSE GOTO falsey0
}
```

```
BLOCK falsey0 {
  part1 <- §SELECT struct_pack(partkey := l.l_partkey, price := l.l_extendedprice * (1 - l.l_discount) * (1 + l.l_tax))
           FROM   lineitem AS l
           WHERE  l.l_orderkey = {0}
           AND    l.l_linenumber = {1}§[orderkey, p1];
  pmargin <- §0§[];
  p2 <- §{0}§[items];
  GOTO inter4
}
```

```
BLOCK inter4 {
  cheapest <- §{0}.price§[part1];
  IF §{0} < 1§[p2]
  THEN GOTO truthy1
  ELSE GOTO falsey1
}
```

```
BLOCK falsey1 {
  part2 <- §SELECT struct_pack(partkey := l.l_partkey, price := l.l_extendedprice * (1 - l.l_discount) * (1 + l.l_tax))
           FROM   lineitem AS l
           WHERE  l.l_orderkey = {0}
           AND    l.l_linenumber = {1}§[orderkey, p2];
  GOTO inter9
}
```

```
BLOCK inter9 {
  IF §{0}.partkey = {1}.partkey§[part1, part2]
  THEN IF §{0}.price < {1}§[part2, cheapest]
       THEN GOTO truthy3
       ELSE GOTO merge2
  ELSE GOTO merge2
}
```

```
BLOCK truthy3 {
  pmargin <- §{0}.price - {1}.price§[part1, part2];
  cheapest <- §{0}.price§[part2];
  GOTO merge2
}
```

```
BLOCK merge2 {
  p2 <- §{0} - 1§[p2];
  JUMP inner_head
}
```

```
BLOCK inner_head {
  IF §{0} < 1§[p2]
  THEN GOTO truthy1
  ELSE GOTO falsey1
}
```

```
BLOCK truthy1 {
  margin <- §{0} + {1}§[margin, pmargin];
  p1 <- §{0} - 1§[p1];
  JUMP outer_head
}
```

```
BLOCK outer_head {
  IF §{0} < 1§[p1]
  THEN GOTO truthy0
  ELSE GOTO falsey0
}
```

```
BLOCK truthy0 {
  EMIT §{0}§[margin];
  STOP
}
```