define: HandleInstForExtraction(*inst*, *alias_graph*, *lock_set*, *key_fields*)

```
1:  alias_graph′ := UpdateAliasGraph(alias_graph, inst);
2:  lock_set′  := UpdateLockSet(inst, lock_set, alias_graph′);
3:  if inst is a write or read operation then
4:    var := GetOperand(inst);
5:    var_node := GetAliasNode(var, alias_graph′);
6:    foreach lock_node in lock_set′ do
7:      <var_field, lock_field> := GetProtectedFieldAccess(
8:                            var_node, lock_node, alias_graph′);
9:      if var_field is not NULL then
10:        insert var_field into key_fields;
11:      end if
12:    end foreach
13:  end if
14:  return < alias_graph′, lock_set′, key_fields>;
```

define: ExtractKeyField ()

```
15:  key_fields := ∅;
16:  foreach func in OS code without a caller function do
17:    foreach code_path in GetCodePath(func) do
18:      alias_graph := ∅;
19:      lock_set := ∅;
20:      foreach inst in GetInstructions(code_path) do
21:        < alias_graph, lock_set, key_fields> :=
22:              HandleInstForExtraction(
23:              inst, alias_graph, lock_set, key_fields);
24:      end foreach
25:    end foreach
26:  end foreach
27:  return key_fields;
```