```cpp
// 15-745 S14 Assignment 3: dominators.cpp
// Group: aebtekar, auc
////////////////////////////////////////////////////////////////////////////////

#include "llvm/IR/Function.h"
#include "llvm/Pass.h"

#include "dataflow.h"

using namespace llvm;

namespace {

// 1-1 mapping between indices and variables
std::vector<std::string> itov;
std::map<Value*, int> vtoi;

Elem dominatorsTransition(BasicBlock* block, Elem elem)
{
  // generate the current block
  int idx = vtoi[block];
  elem[idx] = true;
  return elem;
}

class Dominators : public FunctionPass {
 public:
  static char ID;

  Dominators() : FunctionPass(ID) { }

  virtual bool runOnFunction(Function& F) {
    //ExampleFunctionPrinter(errs(), F);

    itov.clear();
    vtoi.clear();
    // find blocks
    for (ilist_iterator<BasicBlock> BI = F.begin(), BE = F.end(); BI != BE; ++BI)
    {
      std::string name = BI->getName();
      vtoi[BI] = itov.size();
      itov.push_back(name);
    }
    // define lattice and do the analysis
    Lattice lattice(itov, true);
    domForwardSearch(F, &lattice, &dominatorsTransition);

    // Did not modify the incoming Function.
    return false;
  }

  virtual void getAnalysisUsage(AnalysisUsage& AU) const {
    AU.setPreservesCFG();
  }

 private:
};

char Dominators::ID = 0;
RegisterPass<Dominators> X("cd-dominators", "15745 Dominators");

}
```