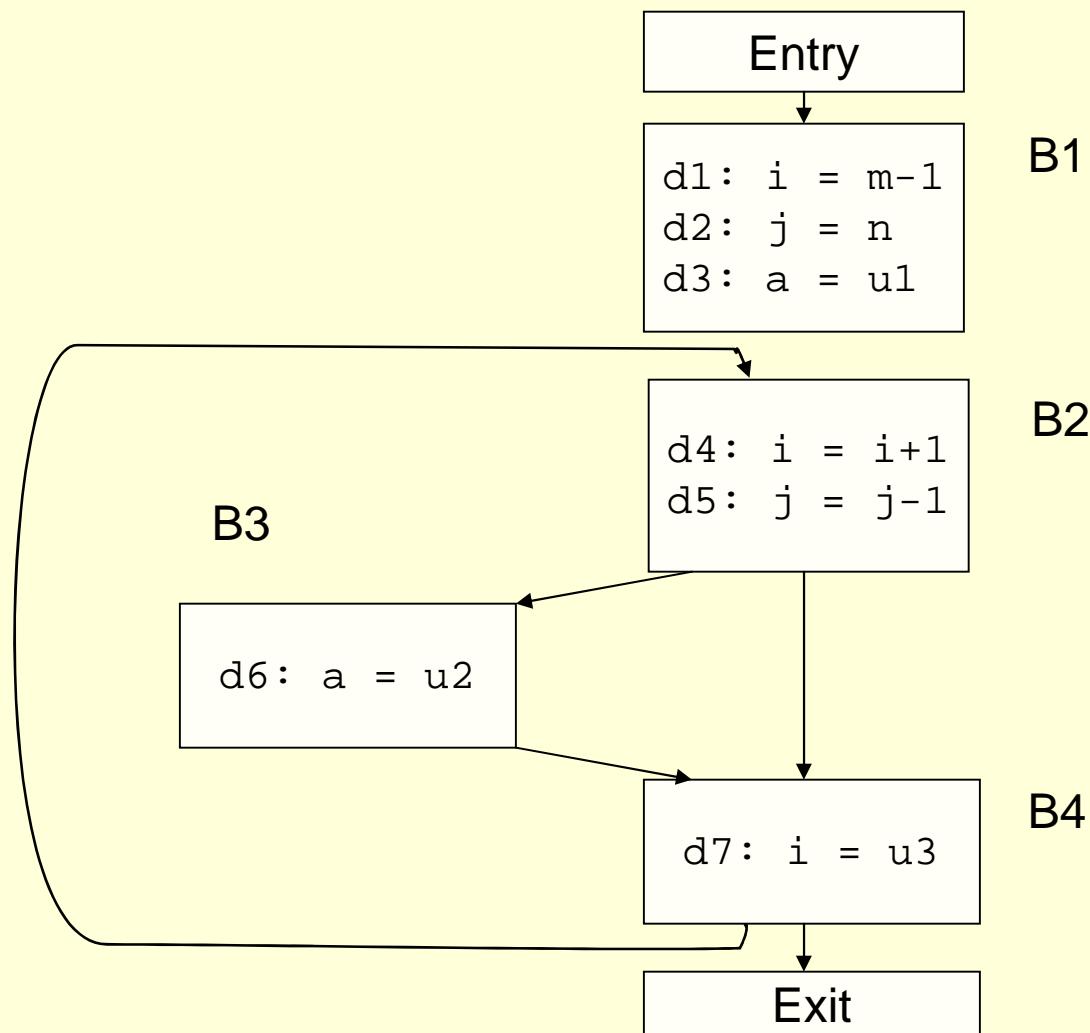


# **Dataflow Analysis**

**(Blackboard Lecture)**

**CS 4447 / CS 9545**  
**Stephen M. Watt**

# A Flow Graph



One entry and one exit for each basic block.

# Dataflow Analysis

- Concerns itself with sets of “properties” that may be true or false at each point in a program.
- E.g.
  - $i^{th}$  variable has been assigned.
  - $j^{th}$  variable has been used.
  - $k^{th}$  definition has been used.

# Dataflow Analysis

- The set of properties that are true before a statement  $s$  is executed is denoted **In[s]**.
- The set of properties that are true after a statement  $s$  is executed is denoted **Out[s]**.
- Dataflow analysis propagates information through the flow graph, either forward (along the arrows), or backward (opposite the arrows).

# Data Flow Analysis

- The execution of statement  $s$  changes which properties are true.

$\text{Out}[s] = f_s(\text{In}[s])$       forward transfer function

$\text{In}[s] = f_s(\text{Out}[s])$       backward transfer function

- For a sequence in a basic block  $B = [s_1; s_2; \dots; s_n]$ , we have

$\text{Out}[s_1] = \text{In}[s_2]$

$\text{Out}[s_2] = \text{In}[s_3]$

...

$\text{Out}[s_{n-1}] = \text{In}[s_n]$

$\text{In}[B] = \text{In}[s_1]$

$\text{Out}[B] = \text{Out}[s_n]$

# Gen and Kill

- The set of properties that are no longer valid after  $B$  *even if they were valid before  $B$*  is said to be “killed by”  $B$  and is denoted  $\text{Kill}[B]$ .
- The set of properties that are valid after  $B$  *even if they were invalid before  $B$*  is said to be “generated by”  $B$  and is denoted  $\text{Gen}[B]$ .
- $\text{Out}[B] = (\text{In}[B] - \text{Kill}[B]) \cup \text{Gen}[B]$

# Examples

- We gave examples of dataflow analysis:
  - Def-Use chains
  - Use-Def chains
  - Copy propagation
  - Dead variable elimination
  - Available expressions
  - Global subexpression elimination