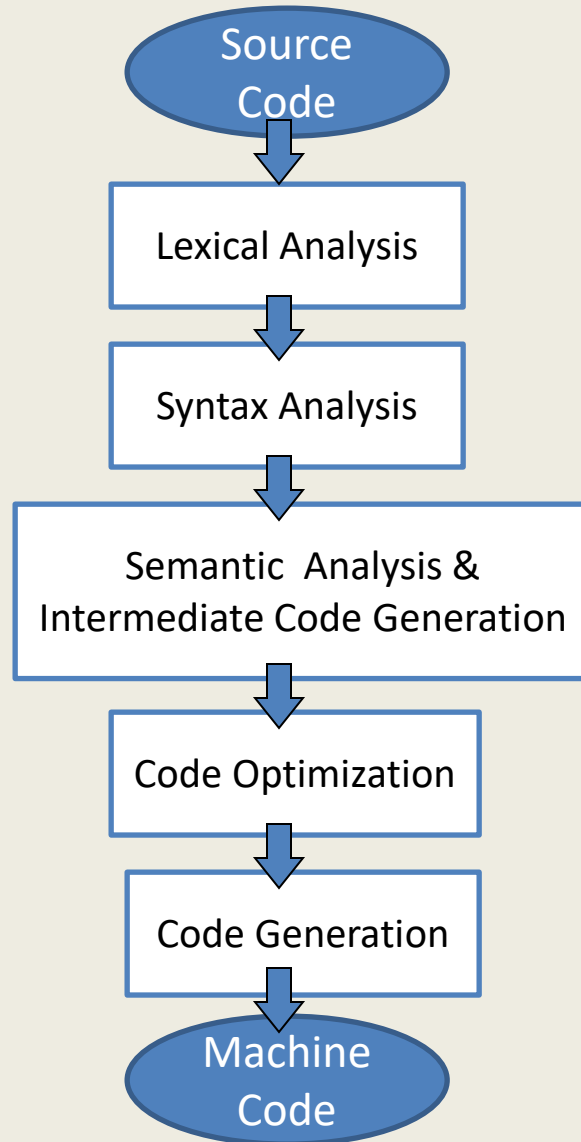


Code Generation

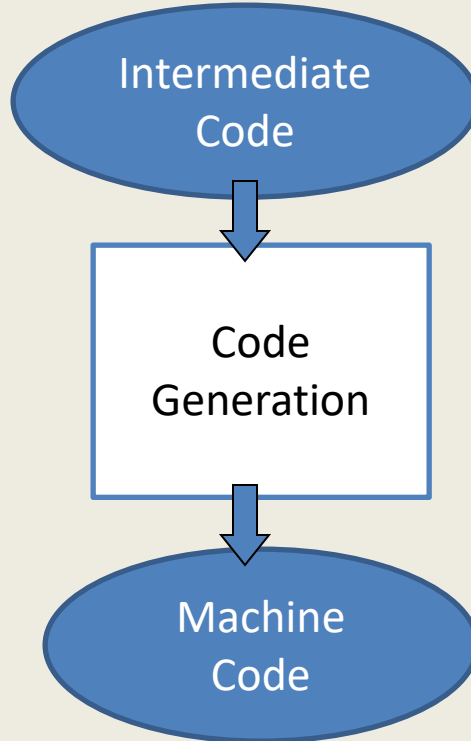
Dinesh Gopalani
dgopalani.cse@mnit.ac.in

Phases in a Compiler

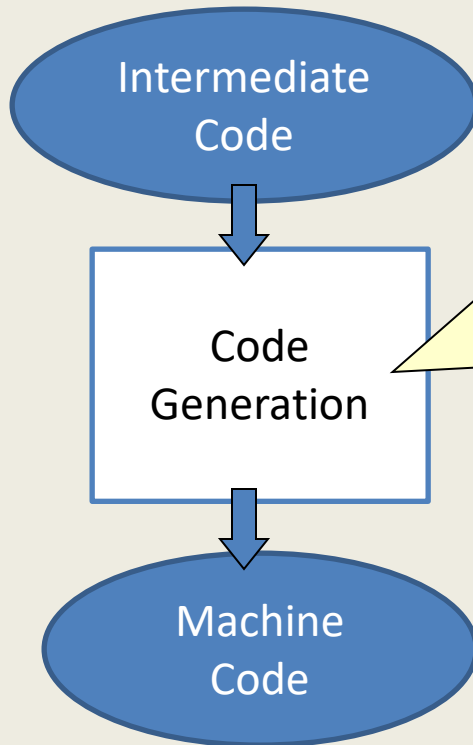


Code Generation

Converts the Intermediate Code into a sequence of Machine Instructions.



How Code Generation works?



There are three primary tasks:

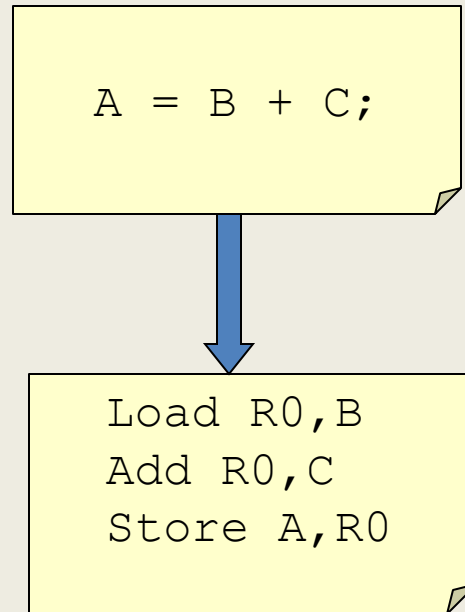
1. Instruction Selection
2. Register Allocation
3. Instruction Ordering

Instruction Selection

- The Intermediate code has to be mapped into a code sequence that can be executed by the target machine.
- The mapping is based on the following:
 - The level of Intermediate Code.
 - The nature of Instruction-set Architecture (RISC vs CISC).
 - The desired quality of the generated code.

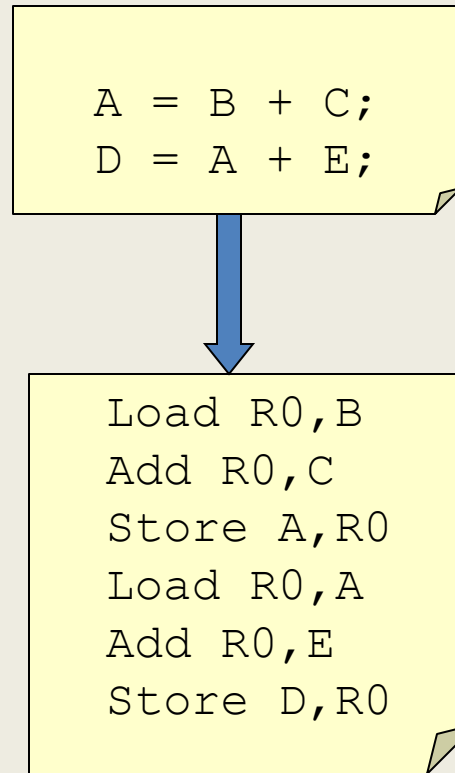
Instruction Selection

Example :



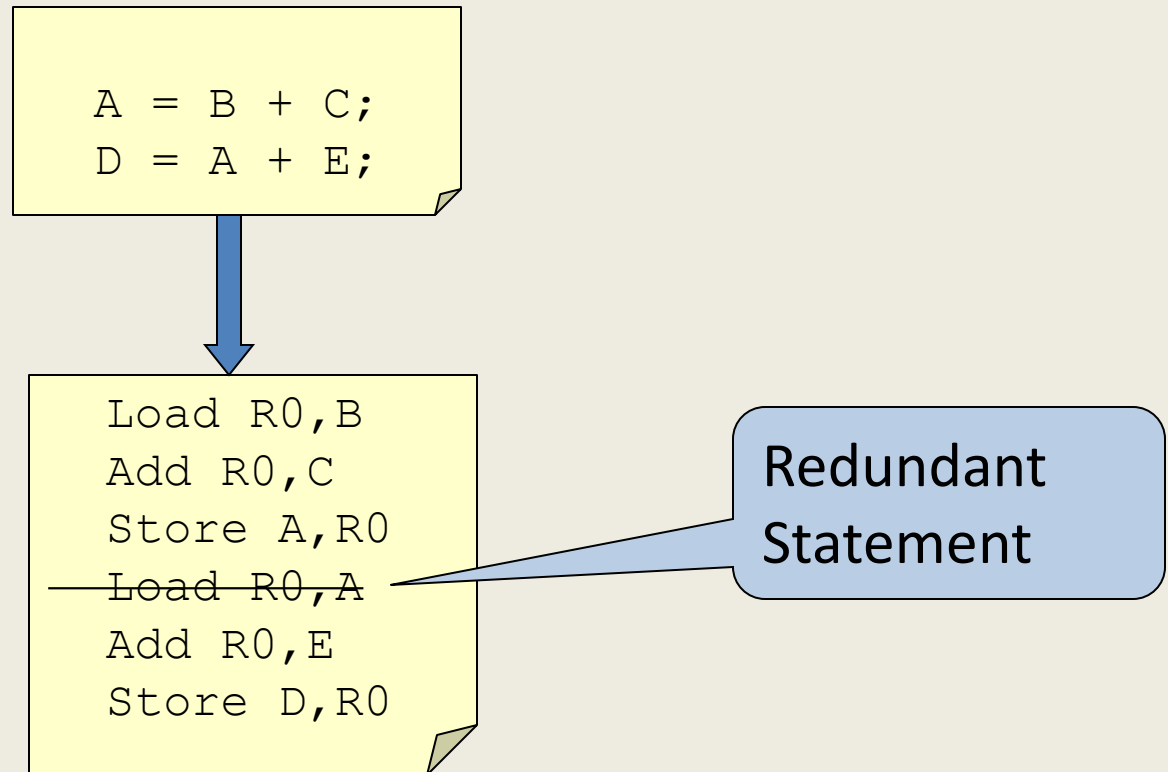
Instruction Selection

Example :



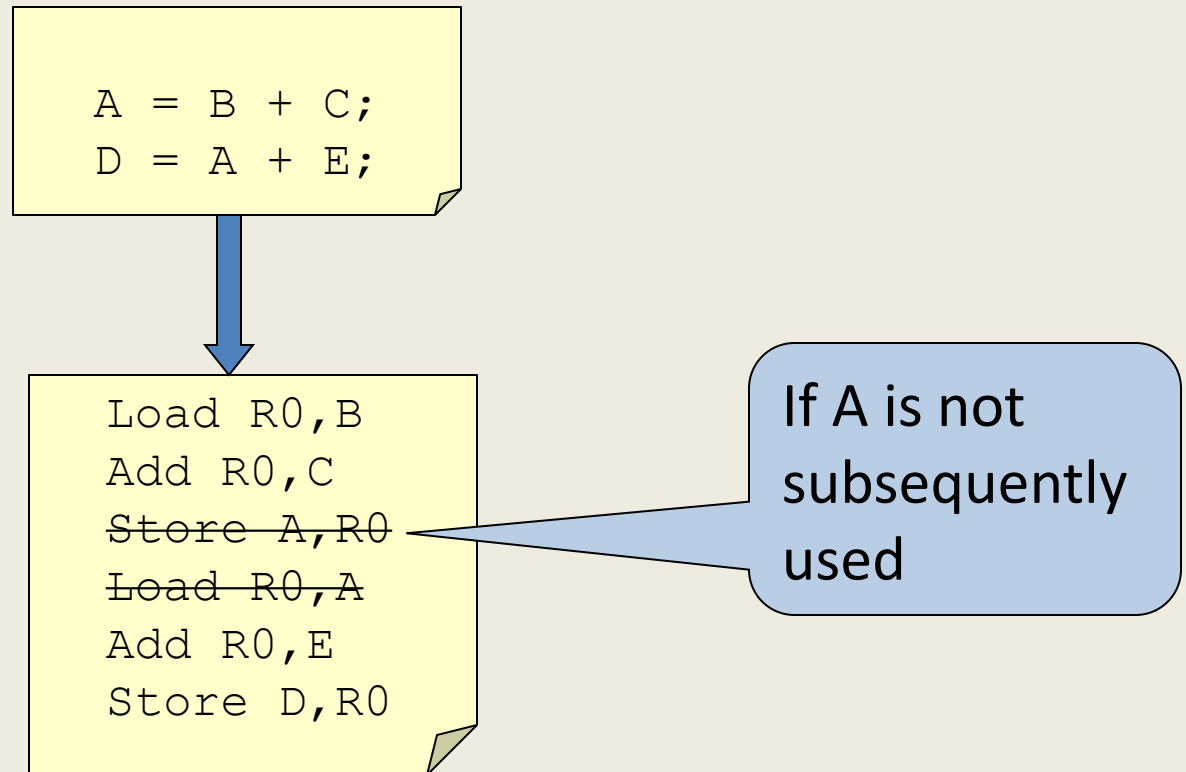
Instruction Selection

Example :



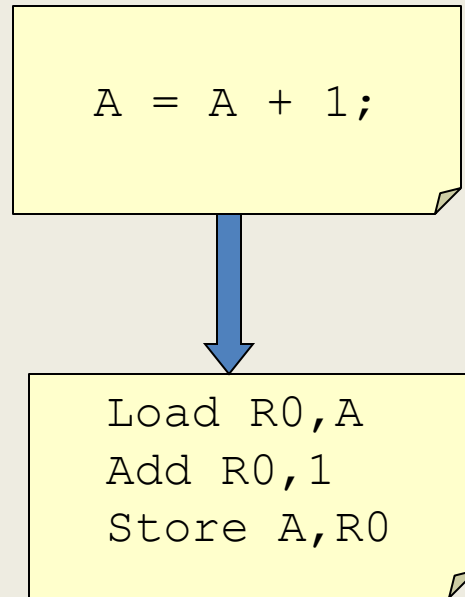
Instruction Selection

Example :



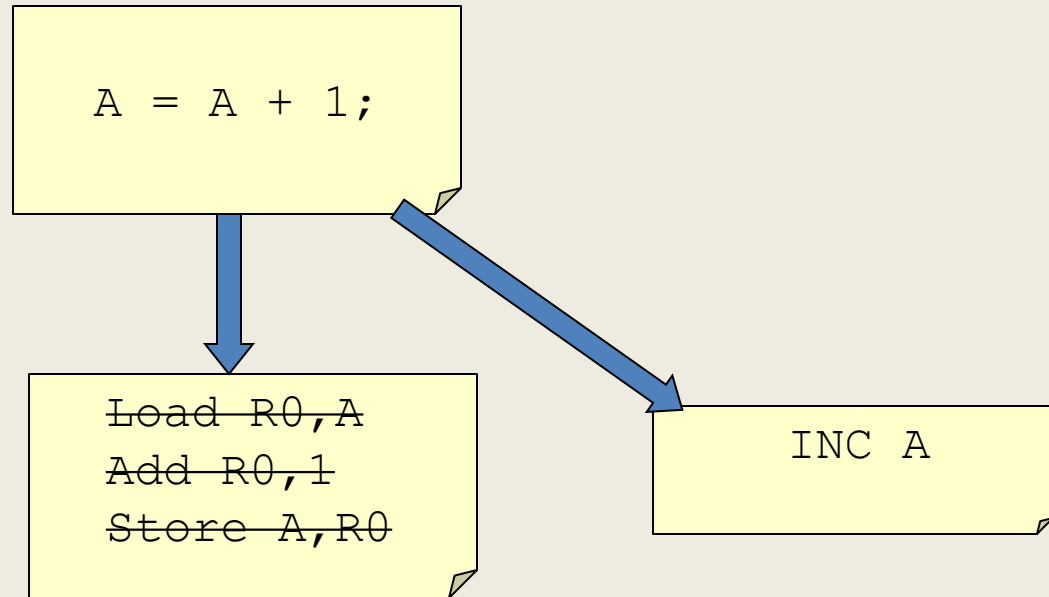
Instruction Selection

Another Example :



Instruction Selection

Another Example :



Register Allocation

- A key problem is deciding what values to hold in what registers.
- Registers are the fastest but not enough to hold all values.
- Instructions involving register operands are shorter and faster than those involving operands in memory.
- So efficient utilization of registers is very important.


Instruction Ordering

- The order in which computations are performed can affect the efficiency of the target code.
- Some computation orders require fewer registers to hold intermediate results than others.
- Finding a best order in the general case is a difficult NP-Complete problem.

Instruction Ordering

Example :

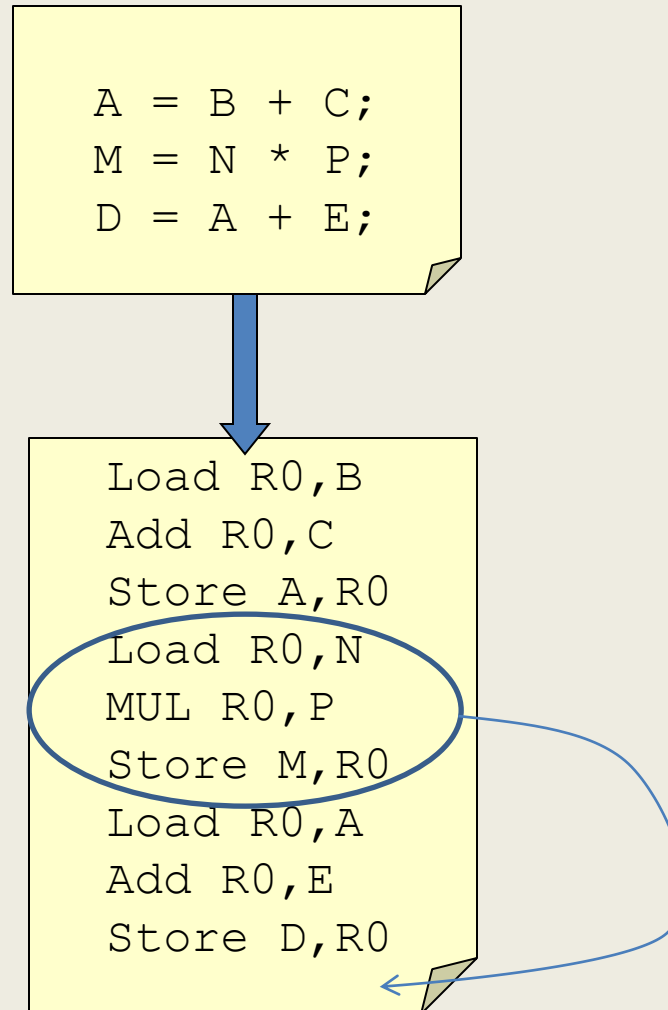
```
A = B + C;  
M = N * P;  
D = A + E;
```



```
Load R0,B  
Add R0,C  
Store A,R0  
Load R0,N  
MUL R0,P  
Store M,R0  
Load R0,A  
Add R0,E  
Store D,R0
```

Instruction Ordering


Example :



Instruction Ordering

Example :

```
A = B + C;  
M = N * P;  
D = A + E;
```




```
Load R0,B  
Add R0,C  
Store A,R0  
Load R0,A  
Add R0,E  
Store D,R0  
Load R0,N  
MUL R0,P  
Store M,R0
```


Instruction Ordering

Example :

```
A = B + C;  
M = N * P;  
D = A + E;
```




```
Load R0,B  
Add R0,C  
Store A,R0  
Load R0,A  
Add R0,E  
Store D,R0  
Load R0,N  
MUL R0,P  
Store M,R0
```

Instruction Ordering

Example :

```
A = B + C;  
M = N * P;  
D = A + E;
```



```
Load R0,B  
Add R0,C  
Store A,R0  
Load R0,A  
Add R0,E  
Store D,R0  
Load R0,N  
MUL R0,P  
Store M,R0
```