

# CS6843: Program Analysis

**Instructor:** Rupesh Nasre (rupesh@cse)

**TAs:** Aman, Manas, Raghesh

Web: ~rupesh/teaching/pa/jan16/  
Moodle: moodle/course/view.php?id=412

Jan 2016

# Examples

End goal	Interesting aspect
Dead code elimination	Reachability
Constant propagation	use-def
Security	Array index range, dangling pointers
Parallelization	Data dependence, SIMD opportunities
Debugging	Slice
Cache performance	Memory access pattern
Memory reduction	Live ranges
...	...

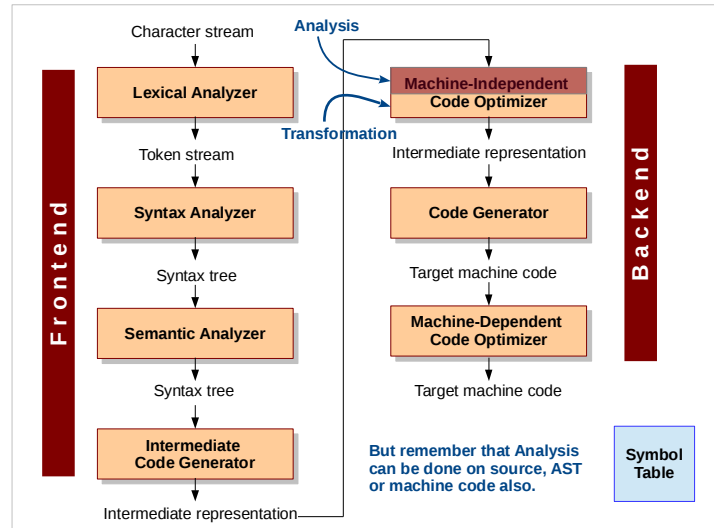
Program Analysis is often a pre-cursor to Optimization.

4

# What is Program Analysis?

For an end-goal identify “interesting aspects” of a program's representation.

2



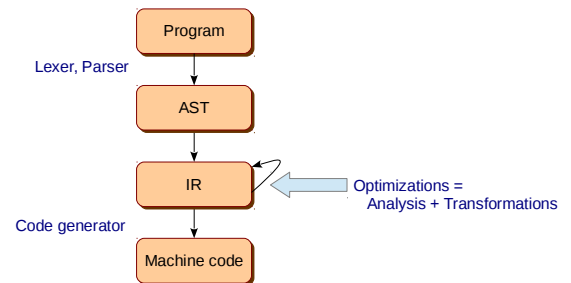
# What is Program Analysis?

For an end-goal identify “interesting aspects” of a program's representation.

- Checking security
- Array index range
- Source, AST, binary, executed instruction

3

# Compiler Organization



6

## Example Three

```
void main() {
  int a, b, c, d, *p;
  p = &a;
  c = a + b;
  d = *p + b;
}
```

Can this computation be avoided?  
(common subexpression elimination)

```
void main() {
  int a, b, c, d, *p;
  p = &a;
  int t = a + b;
  c = t;
  d = t;
}
```

This requires a program analysis called *pointer analysis*.

This requires another analysis called *type analysis*.

7

## In This Course

7. Dynamic Analysis (DYN)
6. Shape Analysis (SHA)
5. Program Slicing (SLI)
4. Parallelization (PAR)
3. Security Analysis (SEC)
2. Pointer Analysis (PTR)
1. Data Flow Analysis (DFA)

10

## Example Two

```
*result = 0;
for (ii = 0; ii < nn; ++ii) {
  strcat(result, str[ii]);
}
```

Can you parallelize this code?

"this" "is" "a" "good" "example" "of" "parallel" "reduction"  
 "thisis" "agood" "exampleof" "parallelreduction"  
 "thisisagood" "exampleofparallelreduction"  
 "thisisagoodexampleofparallelreduction"

Requires *semantic analysis* to figure out that *strcat* performs an **associative** operation.

8

## Logistics

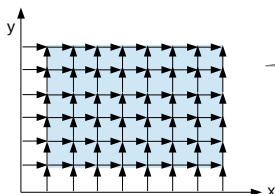
- Moodle for submissions, announcements, discussions
  - Your responsibility to subscribe to it.
- Evaluation:
  - assignments (60%)
  - midsem (20%)
  - endsem (20%)
- C slot (Mon 10, Tue 9, Wed 8, Fri 12).
- Room CS 24.

11

## Example One

```
for (x = 1; x < M; ++x)
  for (y = 1; y < N; ++y)
    a[x, y] = a[x - 1, y] + a[x, y - 1];
```

Can you parallelize iterations?



Requires *loop dependence analysis*

9

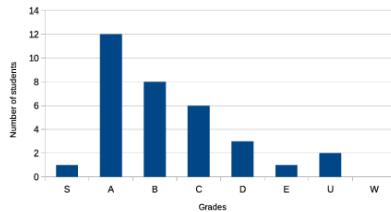
## Assignments

- Four programming assignments (60%).
  - 5 + 15 + 15 + 25
- Assignments would be in LLVM.
- You should work individually.
- You have this week to suggest me any date changes for A2, A3, A4.

12

## Grading

- S  $\geq$  95
- A  $\geq$  80
- B  $\geq$  70
- C  $\geq$  60
- D  $\geq$  50
- E  $\geq$  40



2015 PA evaluation

I don't hesitate to give W grade too.

13

## Course Schedule

Month	Lectures	Evaluations
JAN	DFA, PTA	A1
FEB	SEC	A2, MIDSEM
MAR	PAR, SLI	A3
APR	SHA, DYN	A4
MAY	--	ENDSEM

MidSem and EndSem will have mutually exclusive topics.

14

## CS6843: Program Analysis

**Instructor:** Rupesh Nasre (rupesh@cse)  
**TAs:** Aman, Manas, Raghesh

Web: ~rupesh/teaching/pa/jan16/  
Moodle: moodle/course/view.php?id=412

Jan 2016