CS6843: Program Analysis

Instructor: Rupesh Nasre (rupesh@cse)
TAs: Raghesh, Raja

Web: ~rupesh/teaching/pa/jan14/
Moodle: moodle/course/view.php?id=287

Jan 2014

What *(the hell)* is Program Analysis?

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

What *(the hell)* is Program Analysis?

For an end-goal — Checking security

identify “interesting aspects” — Array index range

of a program's representation. — Source, AST, binary, executed instruction

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.

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*.

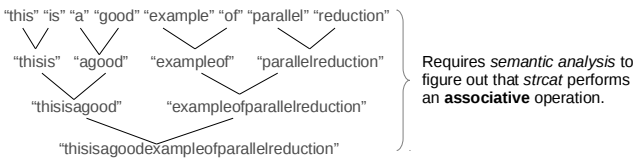
This requires another analysis called *type analysis*.

Example Two

```
*sresult = 0;

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

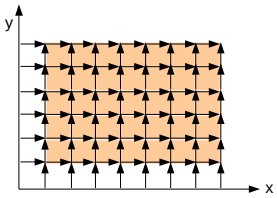
Can you parallelize this code?



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];
```

Requires
loop dependence analysis



More in polyhedral model

In This Course

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

Logistics

- Moodle for submissions, announcements, discussions
- Evaluation:
  - assignments (25%)
  - course project (25%)
  - midsem (25%)
  - endsem (25%)
- R and T slots (Wed, Fri 14:55 – 16:40).
- Room CS 26.
- Assignments would be in LLVM.

Assignments

- Three programming assignments (25%).
- We will increase the complexity (and marks) gradually.
- You can submit late (within two days) but you will lose half marks. Beyond two days, you need not submit.
- You should work individually.
- MidSem will have some questions based on assignments.

Course Project

- 25% marks.
- Need not be in LLVM, need not be in C.
- Sample topics are listed on the webpage, but you can choose your own after discussing with the instructor.
- It will be evaluated in two phases (CP1 and CP2): presentation and demo.
- You can work in a group of one or two.

Course Schedule

Month	Lectures	Evaluations
JAN	DFA, PTR	A1
FEB	SHA, DYN	A2, A3
MAR	POL, PAR	MIDSEM, CP1
APR	SLI, SEC	CP2, ENDSEM

MidSem and EndSem will have mutually exclusive topics.

Course Schedule

Month	Wednesday	Friday	Evaluation	Comments
JAN	15	17	A1: 19	
	22	24		
	29	31		
FEB	5	7	A2: 9	
	12	14		
	19	21	A3: 23	
	26	28		
MAR	5 (MIDSEM)	7	MIDSEM: 5	14:00 – 17:00
	12	14		
	19	21		
	26	28 (CP1)	CP1: 28	In class
APR	2	4		
	9	11 (CP2)	CP2: 11	In class
	16	18		
	23 (ENDSEM)	25 (reserved)	ENDSEM: 23	14:00 – 17:00