Problem Statement:

Recently there has been much talk of multi-threaded computer architectures. However, an open question is, "How much thread parallelism is available in imperative programs?"

Plan:

My 5650 project will attempt to answer the question of how much thread-level parallelism might we be able to exploit in imperative code. To help answer that question, I propose to complete a limit study by measuring the maximum amount of thread parallelism that can be found in a large set of C programs.

Details:

To complete this project I will:

1. Find or build a simple simulator that will "run" generic threaded programs and determine the number of threads executed, the number of instructions executed, and number of cycles required to execute those instructions. From this I can determine the amount of thread-level parallelism exploited during execution.

2. Obtain multiple sets of C benchmarks (from among spec, media, embedded benchmark sets.)

3. Modify an existing compiler to generate threads. For this project each basic block will be considered a thread.

4. Write an experimental report describing my findings.

Deliverables:

Project deliverables will be the software generated and the experimental paper written. The intent will be to produce a paper of publishable quality that can be submitted to a compiler conference such as PLDI or CGO.

Evaluation:

The project will be evaluated based upon the value (as perceived by the grand poobah, Phil Sweany) of the paper written.

This plan is agreed to today:

Date:

Josephine Student:

Phil Sweany:
Problem Statement:

Context-Free grammars were first proposed as a method of performing natural language processing. However, they have been found to lack the semantic power necessary to "solve" that problem.

Plan:

Complete an indepth study of the potential for context-sensitive grammars to be useful within natural language processing.

Details:

To complete this project I will write a survey paper describing research in context-sensitive grammar parsing that focuses on the potential for context-sensitive grammars to be a foundation for natural language processing. This will require a significant literature search to identify

- Parsing techniques, if any, available for context-sensitive grammars
- Use of context-sensitive grammars, if any, in natural language processing

Deliverables:

A survey article describing and evaluating the current state of context-sensitive grammar research, with a particular focus on potential as a formalism for natural language processing.

Evaluation:

The project will be evaluated based upon the value (as perceived by the grand poobah, Phil Sweany) of the paper written.

This plan is agreed to today:

Date:

Josephine Student:

Phil Sweany:
Problem Statement:
It is well known that the ordering of standard optimizations such as
common subexpression elimination, register assignment, and instruction
scheduling have a significant effect on efficiency of compiled code.

Plan:
My 5650 project will attempt to both quantify the variation to be
expected by re-ording intermediate optimizations in a standard C compiler
for a workstation.

Details:
To complete this project I will:
. Find, download, and install a modern C compiler that supports
reordering of intermediate optimizations.

. Find, download, and install a generic mathematical optimization
package such as integer-linear programming, genetic algorithms,
or simulated annealing.

. Obtain multiple sets of C benchmarks (from among spec, media,
embedded benchmark sets.)

. Map the phase-ordering of the intermediate optimizations to
the chosen mathematical optimization.

. Use the mathematical optimization package to find good
ordering(s) for the optimization phases studied.

. Write an experimental report describing my findings.

Deliverables:
An experimental paper describing and analyzing the experiments
completed. The intent will be to produce a paper of publishable quality
that can be submitted to a compiler conference such as PLDI or CGO.

Evaluation:
The project will be evaluated based upon the value (as perceived by
the grand poobah, Phil Sweany) of the paper written.

This plan is agreed to today:

Date:
Josephine Student:
Phil Sweany:
Problem Statement:
IBM's relatively new Cell processor combines computing elements to efficiently execute media and game-programming applications as well as "standard" workstation scalar computation. However, as with most novel parallel architectures, the Cell processor presents daunting compiler challenges.

Plan:
My 5650 project will outline the challenges a compiler for the Cell processor faces, survey available literature for Cell compilers, and design a compiler to support compiler research for Cell compilers.

Details:
To complete this project I will:

- Write a report describing the Cell processor, compiler challenges associated with the Cell processor, the current state-of-the-art in Cell compiler optimization, and a design for building a compiler to do research in Cell compilation.

Deliverables:
An experimental paper describing and analyzing the experiments completed. The intent will be to produce a paper of publishable quality that can be submitted to a compiler conference such as PLDI or CGO.

Evaluation:
The project will be evaluated based upon the value (as perceived by the grand poobah, Phil Sweany) of the paper written.

This plan is agreed to today:

Date:

Josephine Student:

Phil Sweany: