

CS4202-CA – Branch Prediction

This practical is worth **50%** of the practical grade for the modul.. See MMS for submission date.

Main Objectives

- to gain practical experience in simulating aspects of computer architecture;
- to understand the importance (or otherwise) of branch prediction strategies for particular programs;
- to reason about the effect of branch prediction on particular programs;
- to develop and demonstrate scientific experimentation and analysis techniques.

Task

Write a branch predictor simulator which simulates various branch prediction strategies, calculating the misprediction rates.

Select a subset of the provided trace files to test your simulator. You should make a selection of programs, and justify your choices.

You should provide results for Always taken, standard 2-bit predictor, gshare, and a Profiled approach. 2-bit predictor should use a branch predictor table of sizes 512, 1024, 2048 and 4096 entries.

Profiled approach

For the purposes of this practical, a profiled approach is a BP strategy where you may use information gathered from a 'profile' run of the code before it is run or measured in a 'real world' environment. You should think about how such information might be useful, and how it could be represented in architecture.

Be sure to explain clearly how your profiled approach works in the report, including any changes or additions you make to the hardware or toolchain.

Analyse your results and present them in a report. Credit will be given for rigorous scientific experimentation and insightful analysis.

Submit your source code and your report as a PDF. You should also provide a README file to show how to compile and run your program. Trace files can be arbitrarily large. Do not try to analyse huge trace files – you can cut them to a reasonable length. N.B. – when doing this cutting, think carefully about what to cut, and explain your thinking in your report. I have not cut all trace files for you.

You may use any reasonable programming language. If you are unsure what I mean by this, check with me.

Trace files

You have been given a number of trace files for programs. Each file contains the branch trace for a particular program. This is a list of all the branch instructions, which are used in a run of a program and if they were taken or not. Format as follows:

Branch address	Taken
173628492058	1
129576290203	1
028362819233	0
371929471782	0
173628492058	0

Note that the branch at address 173628492058 appears twice. This is dynamic trace of a program, so every time a branch is used, it will appear in the file. Of course, the outcome of the branch can be different. The real files are very long. Branches appear chronologically as they were executed. **N.B. This is a ground truth of what branches were taken, not a prediction.**

You may choose pre-generated trace files from Studres. You must still select the programs used from the subset provided. Not all single programs or combinations are a good choice. You must decide what to use.

Marking

See the standard mark descriptors in the School Student Handbook:

http://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/feedback.html#Mark_Descriptors

You will be assessed on the quality of the report and the design of experiments. Important issues include which benchmarks you choose, how you measure your results, how you display your results, your critical analysis of the results and your explanation of how the experiments were conducted and the rationale behind them.

I am not interested in the quality of the software provided in terms the quality of engineering. The software must simply be able to produce the results reported.

A passing submission will contain the implementation of most of the required strategies, and a report with a basic description of experimental methodology and evidence experimentation being performed by reporting results. A level of basic analysis of the results is expected.

A good submission will contain, in addition, some insight into the results, further than simply a description of the numbers. The methodology will be clear and the experimental design will be sound. The results will be presented

appropriately (this is likely to include graphs, designed and labelled correctly). A reasonable attempt at a profiled prediction strategy is expected. A high quality of writing is expected.

An excellent submission will contain, in addition, an experimental design and description such that a reader could re-implement your experiments purely by using the report. The experimental design will demonstrate an understanding of the task at hand – appropriate benchmarks chosen and sensible measurement strategies which are designed to illuminate the differences in BP strategies. There will be detailed analysis of the results, presented in an appropriate manner.

Lateness

The standard penalty for late submission applies (Scheme B: 1 mark per 8 hour period, or part thereof):

<http://info.cs.st-andrews.ac.uk/student-handbook/learning-teaching/assessment.html#lateness-penalties>

Good Academic Practice

The University policy on Good Academic Practice applies:

<https://www.st-andrews.ac.uk/students/rules/academicpractice/>

Deliverables

As usual submit your **program code** and **report** as **PDF** via MMS. You should also provide a README file to show how to compile and run your program.