Inf3 Computer Architecture

Assignment 1
Understanding Branch Prediction

Issued : 7th February 2013
Due : 14th February 2013 at 4.00pm (at the ITO)

name: Georgi Tsatsev
matriculation number: 1045049

The AlwaysTaken predictor I simulate is implemented by a function that takes its input and then loops through every line of the file if the number in the end is zero then it is a miss if it is one then it is a hit. The function counts the misses then divides them to the sum of the hits and misses and multiplies that to 100 to get a percentage value.

The AlwaysNotTaken predictor is implemented by a function that through every line of the file if the number in the end is zero then it is a hit if it is one then it is a miss. The function counts the misses then divides them to the sum of the hits and misses and multiplies that to 100 to get a percentage value as before.

The ProfileGuided predictor is simulated by first creating a profile for the selected input file which is done i a helper function called "profiler()". It determines how often is each branch is taken or not and increments a corresponding value if its taken or decrements if it is not. The ProfileGuided function then takes this profile and the input file and loops through it again with the profile created. If the value from the profile is higher than zero then it predicts it as taken if not as not taken. If the prediction is wrong we increment miss and if it is correct we increment hit. The return of the function is as before a percentage value of the missprediction.

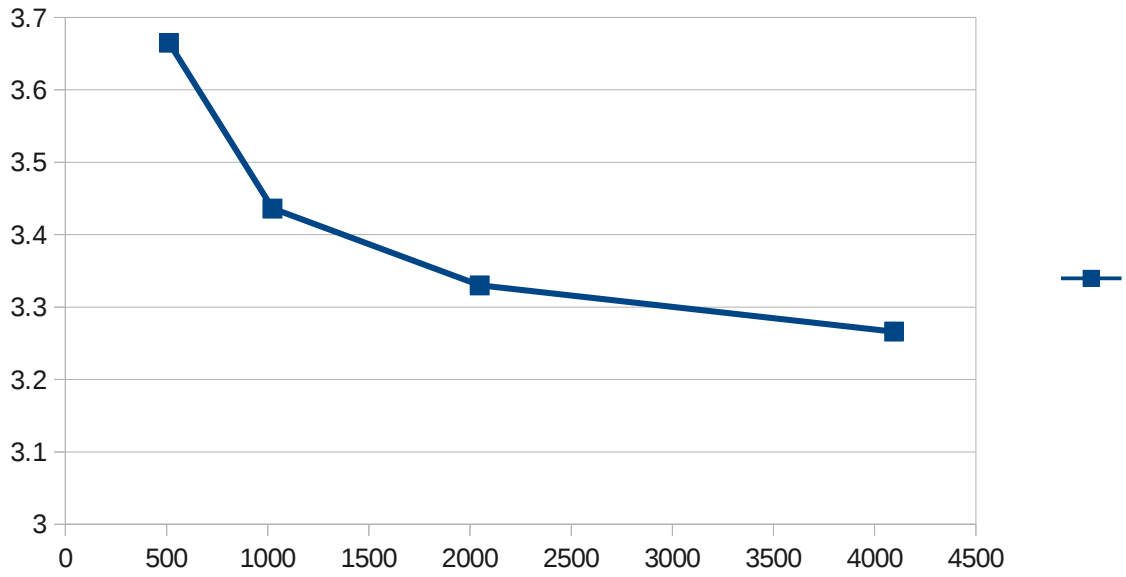
The TwoBitPrediction uses twp helper functions. FromHexToBinary is used to convert the instruction addresses to a binary string that is later used. The fillBinary function is creating the empty prediction table for every possible combination of instructions of length the bits defined. The twobitprediction function is the main function for the prediction and it takes as input a bufferreader and the size of the prediction table. It loops through every line and then with a switch it is determined whether to predict it as taken or not taken and the proper hits and misses are incremented.

The simulator I have created asks for the name of the file and gives options for what type of predictor it would simulate. I tried it with the two files provided and the results for the gcc one were:

```
AlwaysTaken – 23.611%
AlwaysNotTaken – 76.388%
ProfileGuided – 6.628%
TwoBitPrediction with 512 entries table – 3.665%
TwoBitPrediction with 1024 entries table – 3.436%
TwoBitPrediction with 2048 entries table – 3.336%
TwoBitPrediction with 4096 entries table – 3.266%
```

for the mcf:

```
AlwaysTaken –32.240%
AlwaysNotTaken – 67.759%
ProfileGuided – 10.943%
TwoBitPrediction with 512 entries table – 10.155%
TwoBitPrediction with 1024 entries table – 10.155%
TwoBitPrediction with 2048 entries table – 10.155%
TwoBitPrediction with 4096 entries table – 10.155%
```



This is the plot for the twobit prediction where the x values are the prediction table size and y are the misprediction percentage(for gcc). For the mcf one it would not differ since the mispredictions are the same.