

TA: Scott Beamer

Problem 1

Suppose we have made the following measurements:

- Frequency of FP operations = 25%
- Average CPI of FP Operations = 4.0
- Average CPI of other Instructions = 1.33
- Frequency of FPSQR = 2%
- CPI of FPSQR = 20

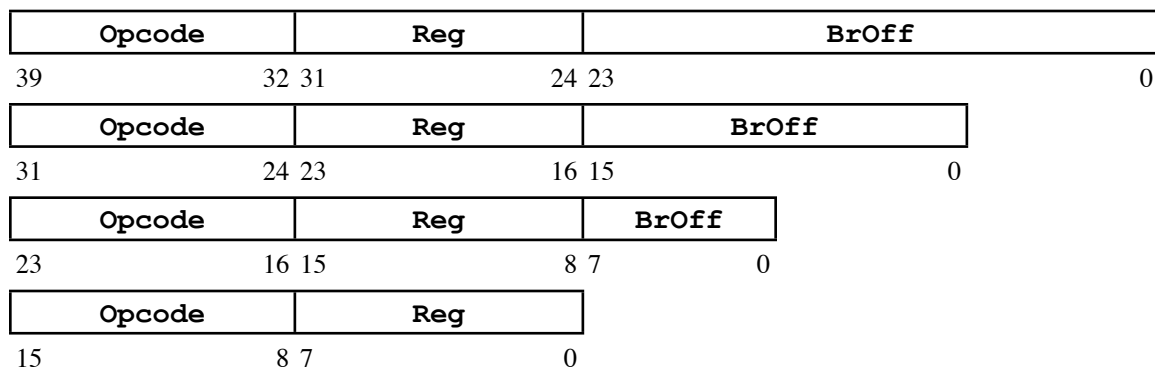
The designers have figured out two design alternatives. One will decrease the CPI of FPSQR to 2 and the other will decrease the average CPI of FP operations to 2.5. Which one will yield the greatest overall improvement in performance? You may assume that FPSQR is not counted as part of FP Operations. Hint: Consider the processor performance equation.

Problem 2

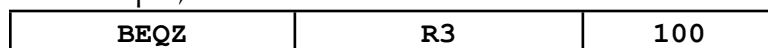
Ben Bitdiddle is designing a handheld device. However, because the device's storage capacity and battery life are limited, he needs to reduce the size of his code. Therefore, he decides to design variable-length instruction set formats for the handheld device to produce more compact code.

Part A

Ben is trying to decide whether it is worthwhile to have multiple offset lengths for branches. He considers the following formats for a branch instruction.



For example,



would mean if **R3** is zero, branch to location **100 + PC**.

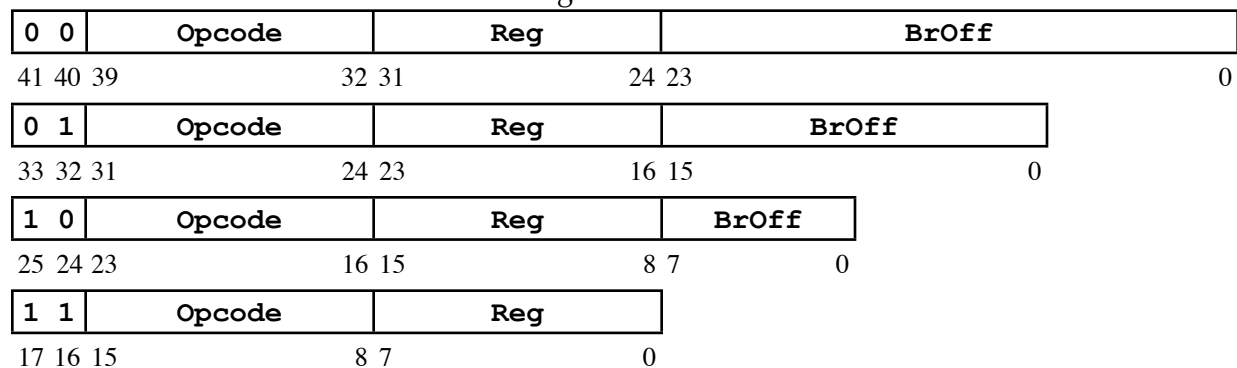
He also has the following statistics reflecting the cumulative percentage of branch instructions that can be accommodated with the corresponding number of bits needed to encode the offset.

# Offset Magnitude Bits	Cumulative Branches	# Offset Magnitude Bits	Cumulative Branches
0	0.10%	11	96%
1	2.80%	12	96.80%
2	10.50%	13	97.40%
3	22.90%	14	98.10%
4	36.50%	15	98.50%
5	57.40%	16	99.50%
6	72.40%	17	99.50%
7	85.20%	18	99.80%
8	90.50%	19	100%
9	93.10%	20	100%
10	95.10%	21	100%

On average, how many bits is a branch instruction reduced by using this variable-length offset encoding, compared with the fixed 24-bit offset? Suppose branch instructions account for 10% of the static code. How much do the variable-sized branch offset encodings reduce the total code?

Part B

In order to implement the above variable-length instruction encoding, Ben needs to specify each instruction's length. Since there are 4 possible instruction lengths, he decides to add 2 more bits as the instruction length field.



What is the overhead cost of adding these 2 bits? Describe two alternative encoding methods to specify the instruction length.

Part C

Ben decides to see if adding extra alignment/decoding hardware to support a smaller granularity of the branch offset field would be worth it. He considers branch offset sizes of 0, 4, 8, 12, 16, 20, 24. By how much does this encoding reduce the total static code (still assuming that branches account for 10% of the code)?

Part D

What are some disadvantages of using variable-length instructions?