```
For the following formulas:
insn = number of instructions
RAW = number of RAW hazards
RAW_1 = number of RAW hazards causing 1-cycle stalls
RAW_2 = number of RAW hazards causing 2-cycle stalls

1. The CPI for the five stage pipeline due to RAW hazards in question 1 is given by the formul
a:
CPI     = (insn + 2*RAW)/insn
The % drop (slowdown) due to RAW hazards can be derived as follows:
% drop  = (CPI_new - CPI_old)/CPI_old * 100
        = ((insn + 2*RAW)/insn - insn/insn)/(insn/insn)
        = 2*RAW/insn

2. The CPI for the five stage pipeline with bypassing due to RAW hazards in question 2 is give
n by the formula:
CPI = (insn + RAW_1 + 2*RAW_2)/insn
The % drop (slowdown) due to RAW hazards can be derived as follows:
% drop  = (CPI_new - CPI_old)/CPI_old * 100
        = ((insn + RAW_1 + 2*RAW_2)/insn - insn/insn)/(insn/insn)
        = (RAW_1 + 2*RAW_2)/insn

The command used to generate to compile the microbenchmark code is 'ssbig-na-sstrix-gcc -O1 mb
q1.c -o mbq1'

The benchmark main loop runs 1 million times. This is enough to effectively eliminate the sign
ificance of the other instructions outside the loop. The loop contains 7 assembly instructions
:
(1) addu    $4,$4,1
(2) addu    $5,$4,1
(3) lw      $8,16($sp)
(4) addi    $7,$8,1
(5) sw      $7,16($sp)
(6) addu    $3,$3,1
(7) slt     $2,$6,$3
(8) beq     $2,$0,$L5

There are 5 RAW hazards in these 8 instructions in the following places:
1. There is a RAW hazard at instruction (2) because it has a dependency on $4. This dependency
 translates to a 2 cycle stall in the pipeline without bypassing and a 1 cycle stall in the pi
peline with bypassing since the the value needed for $4 will be ready by the end of the EXEC s
tage.
2. There is a RAW hazard at instruction (4) because it has a dependency on $8. This dependency
 translates to a 2 cycle stall in both the bypassing and non-bypassing pipelines because the v
alue needed for $8 will be ready at the end of the MEM stage.
3. There is a RAW hazard at instruction (8) because it has a dependency on $2. This dependency
 translates to a 2 cycle stall in the non-bypassing pipeline and a 1 cycle stall in the bypass
ing pipeline because the value needed for $2 will be ready by the end of the EXEC stage.
4. There is a RAW hazard at instruction (5) only in the non-bypassing pipeline. In the bypassi
ng pipeline, even though (5) has a data dependency on $7, there is no hazard because the value
 in $7 is only needed in the MEM stage by instruction (5) and will be ready by the end of the
EXEC stage in instruction (4).
5. There is a RAW hazard at instruction (7) because it has a dependency on $3 which is a 2 cyc
le stall in the non-bypassing pipeline and a 1 cycle stall in the bypassing pipeline.

Therefore, the microbenchmark should generate CPI_1 and CPI_2 as follows:
CPI_1   = (8 + 2*5)/8
        = 18/8
        = 2.25

CPI_2   = (8 + 3 + 2*1)/8
        = 13/8
        = 1.625
```

The actual values measured by running 'sim-safe mbq1' were:

```
CPI_from_RAW_hazard_q1       2.2500 # CPI from RAW hazard (q1)
CPI_from_RAW_hazard_q2       1.6250 # CPI from RAW hazard (q2)
```

which are exactly as predicted.