Question 2 (23 points):

```
# luminosity:
# parameters:
#    $a0: screen address
#    $a1: R (number of rows)
#    $a2: C (number of columns)
.text
luminosity:
    add  $v0, $0, $0      # lumens <- 0
    add  $t0, $0, $0      # i <- 0
next_r:
    add  $t1, $0, $0      # j <- 0
    mul  $t2, $t0, $a2    # $t2 <- i*C
    add  $t2, $a0, $t2    # $t2 <- screen + i*C
next_c:
    add  $t3, $t2, $t1    # $t2 <- screen + i*C + j
    lbu  $t4, 0($t3)      # $t4 <- pixel
    add  $v0, $v0, $t4    # lumens <- lumens + pixel
    addi $t1, $t1, 1      # j <- j+1
    bne  $t1, $a2, next_c # if j < C
    addi $t0, $t0, 1      # i <- i+1
    bne  $t0, $a1, next_r # if i < R
    jr   $ra
```

(a) Version A

```
# lumiptr:
# parameters:
#    $a0: screen address
#    $a1: R (number of rows)
#    $a2: C (number of columns)
lumiptr:
    mul  $t1, $a1, $a2    # $t1 <- R*C
    add  $t1, $a0, $t1    # $t1 <- screen + R*C
    add  $v0, $0, $0      # luminosity <- 0
next_p:
    lbu  $t2, 0($a0)      # $t2 <- pixel
    add  $v0, $v0, $t2    # lumens <- lumens + pixel
    addi $a0,$a0, 1       # p++
    bne  $a0, $t1, next_p
    jr   $ra
```

(b) Version B

Figure 1: Two versions for a luminosity function.

Figure 1(a) above shows the code for the function `luminosity`. Two of the parameters for this function are the number of rows, R, and the number of columns, C. Assume that $R > 0$ and $C > 0$.

1. (**5 points**) Write an algebraic expression in terms of R and C for the total number of instructions executed by `luminosity`.

$$\#\ \text{Instructions}_{\texttt{luminosity}} = 3 + 5 \times R + 5 \times R \times C$$

Instruction Class	Average Cycles
Class A (`add`, `addi`)	1
Class B (`mul`)	10
Class C (`bne`, `jr`)	3
Class D (`lbu`)	5

Table 1: Number of cycles for each instruction class.

2. (**8 points**) The instructions executed by `luminosity` can be divided into four classes according to the average number of clock cycles required to execute each instruction as shown in Table 1. If $R = 100$ and $C = 100$ what is the CPI of `luminosity`?

$$\begin{aligned}
\text{CPI}_{\texttt{luminosity}} &= \frac{\#\text{ cycles}}{\#\text{ Instructions}} \\
\#\text{ cycles} &= 1+1+3+(1+10+1+1+3)\times R+(1+5+1+1+3)\times R\times C \\
\#\text{ cycles} &= 5+16\times R+11\times R\times C \\
\text{CPI}_{\texttt{luminosity}} &= \frac{5+16\times R+11\times R\times C}{4+4\times R+5\times R\times C} \\
\text{CPI}_{\texttt{luminosity}} &= \frac{5+16\times 100+11\times 100\times 100}{4+4\times 100+5\times 100\times 100} = \frac{111605}{50404} = 2.21\frac{\text{cycles}}{\text{instruction}}
\end{aligned} \tag{1}$$

3. (**10 points**) The performance of the system that computes `luminosity` was not satisfactory and the design team decided to improve it. They developed the code `lumiptr` shown in Figure 1(b). The original `lumosity` was running on a system with a 3 GHz clock (1 GHz = 10^9 Hz). They replaced this system with one that runs at 4 GHz. For $R = 100$ and $C = 100$, how much faster, expressed as a ratio, is `lumiptr` in comparison with `luminosity`?

$$\begin{aligned}
\text{Time} &= \frac{\#\text{ ClockCycles}}{\text{Clock Frequency}} \\
\#\text{ ClockCycles}_{\texttt{luminosity}} &= 6+15\times R+11\times R\times C = 111506 \\
\#\text{ ClockCycles}_{\texttt{lumiptr}} &= 10+1+1+3+(5+1+1+3)\times R\times C = 15+10\times R\times C = 100015 \\
\text{Speedup} &= \frac{\text{Time}_{\texttt{luminosity}}}{\text{Time}_{\texttt{lumopt}}} = \frac{\frac{111506}{3\text{GHz}}}{\frac{110606}{4\text{GHz}}} = \frac{111506}{100015}\times\frac{4\text{GHz}}{3\text{GHz}} = 1.49
\end{aligned}$$

Thus, `lumiptr` is almost 1.5 times faster than `lumosity`