

Student Name:
 Student ID:

Section:

(Q1) Give an analysis of the running time for each of the following program fragments?

(a)

```
sum = 0;
for i = 1 to 2 * n
    for j = 1 to i * i
        for k = 1 to j
            if (j mod i = 1)
                sum++;
```

$$T(n) = \sum_{i=1}^{2n} \sum_{j=1}^{i^2} \sum_{k=1}^j \Theta(1) = O(n^5)$$

(b)

```
gcd(a, b) {
    if (a mod b = 0)
        return b
    else
        return gcd(b, a mod b);
}
```

$$T(n) = T\left(\frac{n}{a \bmod b}\right) + \Theta(1), \text{ where } n = ab$$

(c) Assume $m = 2^n$

```
pow = 2
for i = 1 to log m
    pow = pow * pow
return pow
```

$$T(n) = O(n)$$

(Q2) Is $n^2/\log n \in O(n(\log n)^2)$? Show your work

$$\frac{n^2}{\log n} \leq c n(\log n)^2$$

$$\frac{n^2}{\log n} \leq \frac{c n(\log n)^2}{n(\log n)^2}$$

$$\frac{n}{(\log n)^3} \leq c$$

There is NO $c > 0$ where this is always true