Question 5

a) $f(n) = (log_2 n)^2$ and $g(n) = 2log_2 n^{log_2 n} = 2(log_2 n)^2$

According to the definition of asymptotically tight bounds. Let f and g are two functions

$$\lim_{n \to infinite} \frac{f(n)}{g(n)} = constant$$

Then $f(n) = \theta(g(n))$

b) Showing that $n^{10} < c2^{n^{\frac{1}{10}}}$ for some positive number c and n > n0

$f(n) = n^{10}$ and $g(n) = c2^{n^{\frac{1}{10}}}$ taking logarithms for both equations.

$$10 \log n < \log c + \log 2^{n^{\frac{1}{10}}}$$

The equation holds when

$$10 \log n < \log c + n^{\frac{1}{10}}$$

Set c = 1 then the equation would be

$$10 \log n < n^{\frac{1}{10}}$$

Now we apply L'Hopital's rule to proof if the assumption is correct

$$\lim_{n\to\infty} \frac{10 \log n}{n^{\frac{1}{10}}} = \lim_{n\to\infty} \frac{(10 \log n)'}{(n^{\frac{1}{10}})'} = \lim_{n\to\infty} \frac{10\frac{1}{n}}{0.1n^{0.1-1}} = \lim_{n\to\infty} \frac{10\frac{1}{n}}{0.1\frac{1}{n}*n^{0.1}} = \lim_{n\to\infty} \frac{10}{0.1*n^{0.1}} = 0$$

Therefore, $f(n) = O(g(n))$

In addition, there is a fact which indicated that ***for every b > 1 and every x > 0, we have*** $\boldsymbol{log_b n = O(n^x)}$ ***(algorithm design, Jon Kleinberg).*** Thus, when we compare these two equations:

$$\mathrm{f(n)} = 10\log n \; and \; g(n) = n^{\frac{1}{10}} \log 2$$

It could be directly known $f(n) = O(g(n))$.

c) Taking logarithms for both equations. We get:

$$\log g(n) = \log n$$
$$\log f(n) = \log n + (-1)^n \log n$$

For each these two equations, it could be seen that the definition of asymptotic upper bounds is not suit for f(n). There not exist constants c > 0 and n0 < 0 so that for all n >= n0, having T(n) <= c.

Therefore, neither of the two is the asymptotic upper bounds for each other.