1)

a) $(2^n + n^3) \in O(4^n)$

A function $f(n) \in O(g(n))$ iff there exists positive constants c and n_0 such that $f(n) \leq c \cdot g(n)$ whenever $n \geq n_0$

$2^n + n^3 \leq c \cdot 4^n$ $\quad c = 1, n_0 = 2 \quad \Rightarrow \quad 2^n + n^3 \leq 4^n \Rightarrow 4 + 8 \leq 16$

$\Rightarrow 12 \leq 16$ ✓ so this statement is true

b) A function $f(n) \in \Omega(g(n))$ iff there exist positive constants c and n_0 st. $c \cdot g(n) \leq f(n)$ whenever $n \geq n_0$

$c = 3, n_0 = 2$

$c \cdot n \leq \sqrt{10n^2 + 7n + 3} \Rightarrow 3n \leq \sqrt{10n^2 + 7n + 3} \Rightarrow 6 \leq \sqrt{10 \cdot 4 + 14 + 3}$

$\Rightarrow 6 \leq \sqrt{57} \Rightarrow \sqrt{36} \leq \sqrt{57}$ ✓ so this statement is true

c) $n^2 + n \in o(n^2)$

A function $f(n) \in o(g(n))$ iff for every positive constant c, there is a positive integer n_0 such that $f(n) < c \cdot g(n)$ whenever $n \geq n_0$

If there are c and n_0 values that make $n^2 + n = cn^2$ true, then this statement is false.

$n = (c-1)n^2$ $\quad c = 2, n_0 = 1 \quad \Rightarrow \quad 1 = (2-1)1^2 \Rightarrow 1 = 1$ ✓ so this statement is false

d) $3\log_2^2 n \in \Theta(\log_2 n^2)$

$f(n) \in \Theta(g(n))$ iff there exists positive constants c_1, c_2 and n_0 st

$c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$ whenever $n \geq n_0$

$$c_1 \cdot \log_2 n^2 \leq 3 \cdot \log_2^2 n \leq c_2 \cdot \log_2 n^2$$

$$\frac{c_1 \cdot \log_2 n^2}{\log_2 n} \leq \frac{3 \cdot \log_2^2 n}{\log_2 n} \leq \frac{c_2 \cdot \log_2 n^2}{\log_2 n}$$

$$c_1 \cdot 2 \leq 3 \cdot \frac{\log_2^2 n}{\log_2 n} \leq c_2 \cdot 2$$

$$c_1 \leq \frac{3}{2} \frac{\log_2^2 n}{\log_2 n} \leq c_2$$

$c_1 = 1, C_2 = 5, n_0 = 2$

$1 \leq \frac{3}{4} \leq 2$ X so this statement is <u>false</u>

e) $(n^3+1)^6 \in O(n^3)$

$$(n^3+1)^6 \leq c.n^3 \Rightarrow \frac{(n^3+1)^6}{n^3} \leq c$$

This inequality shows that for large values of n $(n \to \infty)$, a constant c cannot satisfy this expression everytime. So this statement is <u>false.</u>

2)

a) $2n\log(n+2)^2 + (n+2)^2 \log \frac{n}{2}$

$4n\log(n+2) + (n+2)^2 \log \frac{n}{2}$

$c_1 \cdot g(n) \leq 4n\log(n+2) + (n+2)^2 \log \frac{n}{2} \leq c_2 \cdot g(n)$

$g(n) = n^2 \log n$

$c_1 \cdot n^2 \log n \leq 4n\log(n+2) + (n+2)^2 \log \frac{n}{2} \leq c_2 \cdot n^2 \log n$

$n_0 = 10, \; c_1 = 1, \; c_2 = 2$

$c_1 \cdot 10^2 \cdot 1 \leq 40 \cdot (1{,}08) + 12^2 (0{,}7) \leq c_2 \cdot 10^2 \cdot 1$

$c_1 \cdot 100 \leq 144 \leq 100 c_2$

$c_1 = 1, \; c_2 = 2$

$100 \leq 144 \leq 200$ ✓ $\Rightarrow$ $\underline{\underline{\Theta(n^2 \log n)}}$

b) $c_1 \cdot g(n) \leq 0{,}001 n^4 + 3n^3 + 1 \leq c_2 \cdot g(n)$

$1000 c_1 \cdot g(n) \leq n^4 + 3000 n^3 + 1000 \leq 1000 c_2 \cdot g(n)$

$1000 c_1 \cdot n^4 \leq n^4 + 3000 n^3 + 1000 \leq 1000 c_2 \cdot n^4$

$n_0 = 1, \; c_1 = 4, \; c_2 = 5$

$4000 n^4 \leq n^4 + 3000 n^3 + 1000 \leq 5000 n^4$

$4000 \leq 4001 \leq 5000 \Rightarrow \underline{\underline{\Theta(n^4)}}$

3)

a) $\lim_{n\to\infty} \frac{n^{\log n}}{n^{1,5}} = \lim_{n\to\infty} n^{\log n - 1,5} = \infty \Rightarrow n^{\log n} > n^{1,5}$

$\lim_{n\to\infty} \frac{\log n}{n^{1,5}} = 0$ since $n^{1,5}$ is a faster growing function compared to $\log n$. $\Rightarrow n^{1,5} > \log n$

$\underline{\underline{n^{\log n} > n^{1,5} > \log n}}$

b) Stirling's formula: $n! \cong \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$ for Ω

$\lim_{n\to\infty} \frac{2^n}{n^2} = \frac{\infty}{\infty} \stackrel{\text{l'hopital}}{\Longrightarrow} \lim_{n\to\infty} \frac{2^n \ln(2)}{2n} = \infty \Rightarrow 2^n > n^2$

$\lim_{n\to\infty} \frac{n!}{2^n} = \lim_{n\to\infty} \frac{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n}{2^n} = \lim_{n\to\infty} \frac{\sqrt{2\pi n}\cdot n^n}{2^n \cdot e^n}$

$= \lim_{n\to\infty} \sqrt{2\pi n}\left(\frac{n}{2e}\right)^n = \infty \Rightarrow n! > 2^n$

$\underline{\underline{n! > 2^n > n^2}}$

c) $\lim_{n\to\infty} \frac{n\log n}{\sqrt{n}} = \lim_{n\to\infty} \sqrt{n}\log n = \infty \Rightarrow \underline{\underline{n\log n > \sqrt{n}}}$

d) $n2^n, 3^n$ $\lim_{n\to\infty} \frac{n2^n}{3^n} = n\left(\frac{2}{3}\right)^n = 0$ $\left(\frac{2}{3}\right)^n$ overpowers since its growing rate is greater and approaches 0. $\Rightarrow \underline{\underline{3^n > n2^n}}$

e) $\lim_{n\to\infty} \frac{\sqrt{n+10}}{n^3} = \lim_{n\to\infty} \frac{n+10}{n^6} = 0$

$\Rightarrow \underline{\underline{n^3 > \sqrt{n+10}}}$

4) Worst case would occur if the program needed to check through all the element combinations and the algorithm returns true.

$i = 0$ to $n-2$

$j = i+1$ to $n-1$

a) The basic operation of this algorithm is comparison. Basic operation is executed everytime the inner for loop is executed.

$$\sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} 1$$

inner loop is executed $(n-1)-(i+1)+1$ times $= n-i-1$

outer loop is executed $\frac{(n-2)(n-1)}{2}$ times

$$i = 0 \text{ to } n-2 \Rightarrow (n-1)+(n-2)+\ldots+(n-n+2-1) = (n-1)+\ldots+1 = \frac{(n-1)(n-1+1)}{2}$$

$$= \frac{n(n-1)}{2}$$

$$\Rightarrow \frac{n^2-n}{2} \in \Theta(n^2)$$

5) The basic operation is multiplication. Basic operation is executed the same times as the innermost loop.

$$\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} 1$$

First loop is executed $n-1+1$ times $= n$

Second loop is executed $n \cdot n = n^2$ times

Innermost loop is executed $n \cdot n^2 = n^3$ times $\Rightarrow \in \Theta(n^3)$

6) algorithm (A [0 ... n-1] , number)

```
for i=0 to n-1 do
    for j=0 to n-1 do
        if (A[i]*A[j] == number)
            print "(" + A[i] + "," + A[j] + ")"
```

This program needs to check every possible pair to see if they satisfy the description, and if they do, it will print the pairs.

Outer loop will be executed n-1 times.

Inner loop will be executed (n-1)(n-1) times

Time complexity is $\Theta((n-1)(n-1)) = \Theta(n^2 + ..) = \underline{\underline{\Theta(n^2)}}$