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

$(2^n + n^3) \leq c.4^n$ $\quad n \geq n_0 > 0$

$c > 0$

$\underline{n=2}$

$2^n + n^3 = 2^2 + 2^3 = 4 + 8 = 12$

$c.4^2 \geq 12$ when $c \geq 1$

$\boxed{c = 1 \quad n_0 = 2}$

$\underline{n=3}$

$2^3 + 3^3 = 8 + 27 = 35$

$4^n = 4^3 = 64$

$\vdots$

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

b) $\sqrt{10n^2+7n+3} \in \Omega(n)$

$\sqrt{10n^2+7n+3} \geq cn$ $\quad c>0$

$\quad n \geq n_0 > 0$

$\sqrt{10n^2+7n^2+3n^2} \geq cn$

$\sqrt{20n^2} \geq cn$

$\sqrt{20}\, n \geq cn$

when $c \leq \sqrt{20} = 2\sqrt{5} > 2$

if $(c=2)$

for all $n_0 \geq 1$

$2\sqrt{5}\, n \geq 2n$

$\boxed{c=2 \quad n_0=1}$

$\boxed{\sqrt{10n^2+7n+3} \in \Omega(n)}$

CamScanner ile tarandı

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

little oh : $f(n) \in o(g(n))$

iff there exists positive constants c and n_0 s.t.
$f(n) < c.g(n)$ when $n \geq n_0$

$n^2+n < c.\ n^2$

$n^2+n < n^2+n^2 = 2n^2$

$2n^2 < c.n^2$

when $c > 2$ and $n_0 \geq 1$

$\boxed{c = 3 \quad n_0 = 1}$

$\boxed{n^2+n \in o(n^2)}$ ✓

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

$$3\log_2(\log_2 n) \geq c.\log_2 n^2$$

$$\log_2(\log n)^3 \geq c.\log_2 n^2$$

$$\frac{\log(\log n)^3}{\log n^2} \geq c$$

$$\log_{n^2}(\log n)^3 \geq c$$

$$\frac{3}{2}\log_n(\log n) \geq c$$

$$\log_n(\log n) \geq \frac{2}{3}c \quad \times$$

there is no c positive constant value which holds for every $n \geq n_0$

$$3\log_2^2 n \notin \Theta(\log_2 n^2)$$

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

$(n^3+1)^6 \leq c \cdot n^3$

$n^3+n^3 \geq n^3+1$ when $n \geq 1$

$(n^3+n^3)^6 \leq c \cdot n^3$

$(2n^3)^6 \leq c \cdot n^3$

$2^6 n^{18} \leq c \cdot n^3$

$\boxed{2^6 \cdot n^{15} \leq c}$

there is no constant c and n_0 which satisfy inequality for every $n \geq n_0$

$\boxed{(n^3+1)^6 \notin O(n^3)}$

2)

a) $\underbrace{2n \log (n+2)^2}_{\Theta(f(n))} + \underbrace{(n+2)^2 \log \frac{n}{2}}_{\Theta(h(n))} \in \Theta(g(n))$

$$\underline{2n \log (n+2)^2}$$

$2n \log (n+2)^2 \leq c \cdot f(n)$

$2n \log (n+n)^2 \leq c \cdot f(n)$

$2n \log (2n)^2 \leq c \cdot f(n)$

$2n \log (2n)$

$2n (\log 2 + \log n) \leq c \cdot f(n)$

when $n \geq 2$
$\log n \geq \log 2$

$2n (\log n + \log n) \leq c \cdot f(n)$

$4n \log n \leq c \cdot f(n)$

$\boxed{c \geq 5 \quad f(n) = n \log n \quad n > 1}$ $\boxed{2n \log (n+2)^2 \in O(n \log n)}$

$2n \log (n+2)^2 \geq c \cdot (n \log n)$

$\boxed{2n \log (n+2)^2 > 2n \log(n) \text{ when } n \geq 1}$

$2n \log(n) \geq c \cdot (n \log n)$

$\boxed{c = 1 \quad n = 2}$ $\boxed{2n \log (n+2)^2 \in \Omega(n \log n)}$ 2

$\boxed{2n \log (n+2)^2 \in \Theta(n \log n)}$

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

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

$$(n+2)^2 \cdot (\log n - \log 2) \leq c \cdot h(n)$$

$$(n+2)^2 \cdot (\log n - 1) \leq c \cdot h(n)$$

$$\boxed{\log n - 1 < \log n \quad \text{when } n \geq 1}$$

$$(n+2)^2 \cdot \log n \leq c \cdot h(n)$$

$$n+2 \leq n+n = 2n \quad \text{when } n \geq 2$$

$$(n+n)^2 \cdot \log n \leq c \cdot h(n)$$

$$4n^2 \log n \leq c \cdot h(n)$$

$$\boxed{c \geq 4 \quad n_0 \geq 2}$$

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

$$\boxed{(n+2)^2 \log \frac{n}{2} \in O(n^2 \log n)}$$

$$(n+2)^2 \log \frac{n}{2} \geq c_0 (n^2 \log n)$$

$$n^2 \log \frac{n}{2} \geq c_0 \cancel{n^2} \log n$$

$$\log \frac{n}{2} \geq c \log n$$

$$\log n - \log 2 \geq c \log n$$

$$\log n - 1 \geq c \log n$$

$$\boxed{\begin{aligned} c &= 0.5 \\ n_0 &\geq 4 \end{aligned}}$$

$$\boxed{(n+2)^2 \log \frac{n}{2} \in \Omega(n^2 \log n)}$$

and we also proved that

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

so

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

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

$$\boxed{2n \log (n+2)^2 + (n+2)^2 \log \frac{n}{2} \in \Theta(n^2 \log n)}$$

2b) $0.001n^4 + 3n^3 + 1$

$c_2 \cdot g(n) \leq 0.001n^4 + 3n^3 + 1 \leq c_1 \cdot g(n)$

$\hookrightarrow 0.001n^4 + 3n^3 + 1 \geq c_2 \, g(n)$

$g(n) = n^4 \quad c = 0.001$

$0.001n^4 + 3n^3 + 1 \geq 0.001 \cdot n^4$

$\boxed{3n^3 + 1 \geq 0 \quad \text{when } n \geq 0}$

$\boxed{c = 0.001 \quad n_0 \geq 0}$

$0.001n^4 + 3n^3 + 1 \in \Omega(n^4)$

$0.001n^4 + 3n^3 + 1 \leq c_1 \, g(n)$

$0.001n^4 + 3n^3 + 1 \leq c_1 \, n^4$

$\boxed{0.001n^4 + 3n^4 + n^4 = 4.001n^4 \geq 0.001n^4 + 3n^3 + 1}$ when $n \geq 0$

if $c_1 = 5$

wherever $n \geq n_0 = 1$

$0.001n^4 + 3n^3 + 1 \in O(n^4) \longleftarrow \boxed{0.001n^4 + 3n^3 + 1 \leq c_1 n^4}$

$\boxed{0.001n^4 + 3n^3 + 1 \in \Theta(n^4)}$

3) a) $\log n$, $n^{\log n}$, $n^{1.5}$ $n^{\log n} > n^{1.5} > \log n$

$$\lim_{n\to\infty} \frac{n^{\log n}}{n^{1.5}} = \lim_{n\to\infty} n^{\log n - 1.5} = \infty \qquad n^{\log n} > n^{1.5}$$

$$\boxed{\infty^{\infty - 1.5} = \infty^{\infty} = \infty}$$

$$\lim_{n\to\infty} \frac{\log n}{n^{1.5}} = \frac{\infty}{\infty} \Rightarrow \lim_{n\to\infty} \frac{\frac{1}{n.\ln 2}}{1.5.\frac{1}{\sqrt{n}}} = \lim_{n\to\infty} \frac{1}{n\ln 2} \cdot \frac{\sqrt{n}}{1.5} = \frac{1}{1.5 \times \ln 2 \times \sqrt{n}} = 0$$

$$\boxed{n^{1.5} > \log n}$$

b) $n!,\ 2^n,\ n^2$

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

$$\lim_{n\to\infty} \frac{2^n}{n^2} = \frac{\infty}{\infty} \overset{\text{l'hopital}}{\Longrightarrow} \lim_{n\to\infty} \frac{2^n \cdot \ln 2 \cdot 1}{2n} = \frac{\infty}{\infty} \overset{\text{l'hopital}}{\Longrightarrow} \ln 2 \cdot \lim_{n\to\infty} \frac{2^n \cdot \ln 2 \cdot 1}{2} = \infty$$

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

c) $n \log n$, $\sqrt{n}$

$$\lim_{n\to\infty} \frac{n\log n}{\sqrt{n}} = \frac{\infty}{\infty}$$

$$\lim_{n\to\infty} \frac{\log n + n \cdot \frac{1}{n \ln 2}}{\frac{1}{2}\cdot n^{-\frac{1}{2}}} = \lim_{n\to\infty} \frac{\log n + \frac{1}{\ln 2}}{\frac{1}{2}\cdot n^{-\frac{1}{2}}} = \lim_{n\to\infty} \frac{n^{\frac{1}{2}}\left(\log n + \frac{1}{\ln 2}\right)}{\frac{1}{2}} = \infty \quad \boxed{n\log n > \sqrt{n}}$$

CamScanner ile tarandı

D) $n2^n$, 3^n $\qquad$ $3^n > n2^n$

$$\lim_{n\to\infty} \frac{n2^n}{3^n} = \frac{\infty}{\infty} \Rightarrow \lim_{n\to\infty} \frac{1\cdot 2^n + n\cdot(2^n\cdot \ln 2)}{3^n\cdot \ln 3} = \lim_{n\to\infty} \frac{2^n(1+n\ln 2)}{3^n \ln 3} = \lim_{n\to\infty} \frac{(1+n\ln 2)'}{\left(\left(\frac{3}{2}\right)^n\cdot \ln 3\right)'}$$

$$\lim_{n\to\infty} \frac{\ln 2}{\left(\frac{3}{2}\right)^n\cdot \ln\frac{3}{2}\cdot \ln 3} = 0$$

e) $\sqrt{n+10}$, n^3

$$\lim_{n\to\infty} \frac{\sqrt{n+10}}{n^3} = \frac{\infty}{\infty} \Rightarrow \lim_{n\to\infty} \frac{\frac{1}{2}\cdot(n+10)^{-1/2}}{n^3} = \frac{\frac{1}{2}}{n^3\cdot(n+10)^{1/2}} = \frac{1/2}{\infty} = 0$$

$$\boxed{n^3 > \sqrt{n+10}}$$

4) a) This algorithm checks if the given two dimensional array is symmetric or not. It returns true if the given array is symmetric.

Worst case happens when the given array is symmetric.

It's basic operation is comparison: B[i,j] != B[j,i]

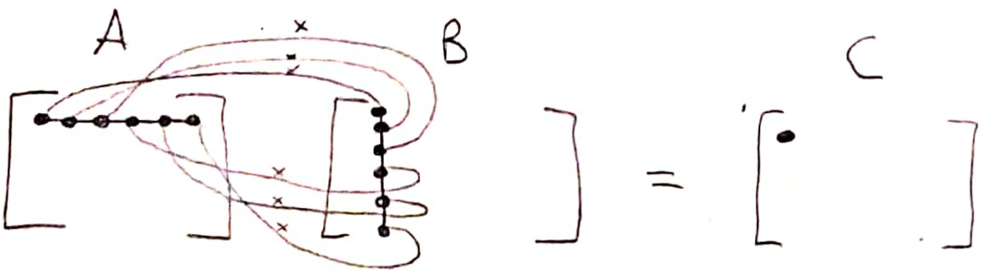
b) $\sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} 1$

c)

$i=0 \Rightarrow n-1$ step
$i=1 \Rightarrow n-2$ step.
$\vdots \qquad \vdots$
$i=n-3 \Rightarrow 1$ step
$i=n-2 \Rightarrow 0$ step

$$\boxed{\frac{(n-1).(n)}{2} = \frac{n^2-n}{2} \in \Theta(n^2)}$$

5)



a) It multiplies 2 matrices and returns the answer of the multiplication.

It's basic operations are multiplication, addition and assignment.

C[i,j] = C[i,j] + A[i,k] x B[k,j]

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

c) $n \times n \times n = n^3 \in \Theta(n^3)$

```
Find Multiplication ( A[0..n-1], double Mult)
    Create a empty hashmap which holds double for key and List for value
    for ( int i = 0 to A.length)
        List list = Map.get( A[i])
        if ( list is null)
            Map.put( A[i], new List());
        Map.get( A[i]).add(i);

    for( i = 0 to A.length)
        if (mult is zero)
            if( A[i] != 0)
                Searched = 0
            else
                Searched = A[i]
        else
            if( A[i] is equal to 0)
                Continue
            Searched = mult/A[i]

        List list = map.get (searched)
        if ( list is not null)
            for( int j in list) // iterator
                if(( mult is zero AND j != i AND A[i] != A[j])
                                OR
                               j > i)
                    print (arr[i] + "," + arr[j])
```

Worst case occurs when all list elements are square root of the given element. Because searched element is equal to the all elements and when list is taken from map it contains all elements. So that, all list is traveled for all of the list elements.

$$\sum_{i=0}^{n} \sum_{j=0}^{n} 1 = n \times n = n^2 \in \Theta(n^2) = w$$

Best case happens when the array does not contains wanted pairs. When this is the case, there are two non-nested loops.

$$\sum_{i=0}^{n} 1 + \sum_{i=0}^{n} 1 = n + n = 2n \in \Theta(n) = \text{best case}$$

If the numbers are random, it is very likely to have cases which are closer to best case. will not happen generally.

```
import java.util.ArrayList;
import java.util.HashMap;
import java.util.HashSet;
import java.util.LinkedList;

public class Odev {
private static int counter=0;
public static void main(String[] args) {
double[] arr = {3,2,6,2,1,4,6,3};
multiplicationPairs(arr,6);
}
private static void multiplicationPairs(double [] arr,double mult){
HashMap<Double,LinkedList<Integer>> set = new
HashMap<Double,LinkedList<Integer>>();
for(int i=0;i<arr.length;i++) {
LinkedList<Integer> list = set.get(arr[i]);
if(list == null)
set.put(arr[i], new LinkedList<Integer>() );
set.get(arr[i]).add(i);
}
double searched;
for(int i=0;i<arr.length;i++) {
if(mult == 0) {
if(arr[i] != 0) {
searched = 0;
} else {
searched = arr[i];
}
} else {
if(arr[i] == 0)
continue;
searched = mult/arr[i];
}
LinkedList<Integer> list = set.get(searched);
if(list != null) {
for(int j:list) {
if((mult == 0 && j != i && arr[i] != arr[j]) || (j>i))
System.out.printf("(%f,%f)\n",arr[i],arr[j]);
}
}
}

}
}
```