Pb. 1 (a) Build a minheap with k elements
for $i = k+1$ to n:
insert a new element
remove min element.

Comparisons: $(n-k)\lg k \approx n \lg k$
(more solutions from Midterm II practice).

(b) $(k!)^{n/k} \leq$ number of leaves $\leq 2^h$

$$2^h \geq (k!)^{n/k}$$

$$h \geq \frac{n}{k} \lg k!$$

$\lg\left(\sqrt{2\pi k}\left(\frac{k}{e}\right)^k\right)$
$= \frac{1}{2}\lg(2\pi k) + k\lg k - k\lg e$
$\approx \frac{n}{k} \cdot k\lg k = n\lg k$

$$= \frac{n}{k}\left[\lg(k\cdot(k-1)\cdots 2\cdot 1)\right]$$

$$= \frac{n}{k}\left[\lg k + \lg(k-1) + \cdots + \lg 2\right]$$

$$= \frac{n}{k}\sum_{i=2}^{k} \lg i$$

$$= \frac{n}{k}\int_1^k \lg x\, dx = \frac{n}{k\ln 2}\int_1^k \ln x\, dx \qquad \lg x = \frac{\ln x}{\ln 2}$$

$$\int \ln x = \ln x \int dx - \int \frac{d}{dx}\ln x \int dx$$

$$= x\ln x - \int \frac{1}{x}\cdot x\, dx = x\ln x - x \Big|_1^k = k\ln k - k$$

$\Rightarrow h \geq n/k\ln 2\,(k\ln k - k) \approx n\lg k$ (higher order term)

Pb 2 (a)

```
find kthElement (A, k)
    index = findN3rd(A)
    if A[k] == A[index]:
        return A[k]
    else if A[index] > A[k]:
        findkthElement(A[1:n/3], k)
    else
        findkthElement(A[n/3:n], k - n/3)
```

2(b) $T = T\left(\frac{2n}{3}\right) + 2n + n - 1$

2(c) Guess: $T(n) \leq a\,n$ $\qquad T(1) = 9.$

$$T(n) \leq a\frac{2n}{3} + 2n + n - 1$$

$$= a\frac{2n}{3} + 3n - 1$$

$$= \left(\frac{2a}{3} + 3\right)n - 1$$

$$\leq an$$

$$\Rightarrow \frac{2a}{3} + 3 = a$$

$$\Rightarrow 3 = \frac{a}{3}$$

$$\Rightarrow a = 9$$

$T(n) \approx 9n \quad ; \quad T(1) = 9$

(2d) There are the following possibilities:

① They end up in different partitions

② They end up in the same partition

As a result, in the worst case

$$T(n) \leq T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + 3n - 1$$

Now run your algorithm of the partition $T(n/3)$ and separately on the partition $T(2n/3)$

(2e) $T(n) \leq an$; with $a = 9$

$$T(n) \leq T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + 3n - 1$$

$$= \frac{an}{3} + \frac{2an}{3} + 3n - 1$$

$$= \frac{9n}{3} + \frac{2 \times 9n}{3} + 3n - 1$$

$$= 12n - 1$$

Pb. 3) (a) 1, 2

(b) $n \geq 1$

(c) $n \geq 1$

(Pb. 4)

```
A = ∅
numtriangle = 0
for i = 1 to n:
  for j = i+1 to n:
    if A[i,j] == 1
      for k = j+1 to n:
        if A[i,k] == 1 && A[j,k] == 1:
          numtriangle += 1
          A = A ∪ [i,j,k]
```

$O(m)$

$O(n)$

$O(m \cdot n)$

```
for each vertex v ∈ V:
  for u, w in Adjlist(V):
    if deg(u) > deg(v) and deg(w) > deg(v):
      if u,v ∈ E and v,w ∈ E:
        A = A ∪ [u, v, w]
        numtriangle += 1
```

$O(m \cdot \sqrt{m}) \approx O(m^{3/2})$

Degrees of all vertices can be computed in $O(m+n)$

$deg(v) \leq \sqrt{m}$

$\sum deg\, v \leq 2m$

Pb 5 If the clause contains 2 literals then create two AND'ed clauses

$(x_1 \vee x_2)$

replace it with

$$(x_1 \vee x_2 \vee l) \wedge (x_1 \vee x_2 \vee \bar{l})$$

If the clause contains 1 literal then create four AND'ed clauses

(x_1)

replace it with

$$(x_1 \vee l_1 \vee l_2) \wedge (x_1 \vee \bar{l}_1 \vee l_2)$$
$$\wedge\ (x_1 \vee l_1 \vee \bar{l}_2) \wedge (x_1 \vee \bar{l}_1 \vee \bar{l}_2)$$

(P6) 6a) Given a graph $G = (V, E)$ with integer weights on the vertices and a target T, is there a vertex cover whose sum of weights on the vertices is at most T

6b) $A[i, j] \leftarrow$ Adj. matrix

$w[i] \leftarrow$ weight of vertex i

Certificate: set of vertices in the weighted vertex cover.

It can be represented by an array of size n where incover[i] is TRUE if vertex i is in the vertex cover

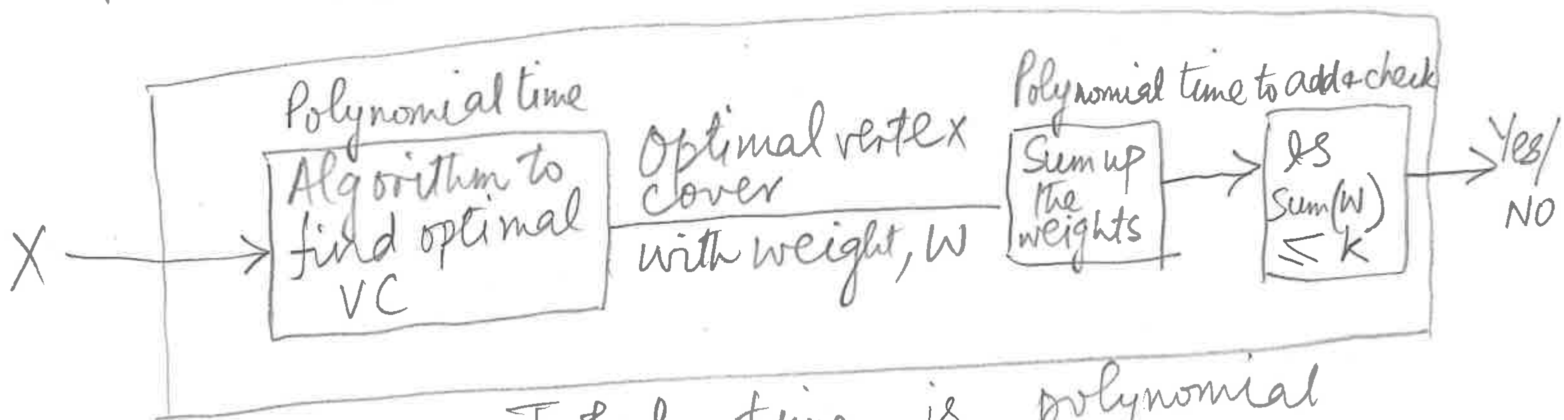
Check Vertex Cover

$O(n)$ to check vertex cover is atmost Target T and $O(n^2)$ to check every edge

$O(n^2)$

```
W ← 0
for i = 1 to n:
    if incover[i] then W ← W + w[i]
if W > T then return FALSE
{check every edge is covered}
for i = 1 to n
    for j = i+1 to n
        if A[i,j] and NOT (incover[i] or incover[j])
            return FALSE
return TRUE
```

Pb 6(C) Optimization to decision



Total time is polynomial

Pb 6(d) A vertex cover of total weight $\leq W$ exists

$W = 0$, $k = 0$

while notVertexCover(W, k, G):

  W += 1

  k += 1

for $i = 1$ to n:

  remove vertex i

  $W' = W - W_i$

  if decisionVersion(W') == Yes:

    $W \leftarrow W'$

  else:

    put vertex back in vertex cover