1.13

a) 17 is $O(1)$

17 is a constent therefore it is $O(k)$ where $k \geq 1$

b). $\frac{n(n-1)}{2}$ is $O(n^2)$

$= \frac{n^2 - n}{2}$ the highest polynomial is $n^2$

therefore it is $O(n^2)$

c). $\max(n^3, 10n^2)$ is $O(n^3)$ $n^3 \leq cn^3$ where $c \geq 1$

~~$T(n) \leq cn^2$~~, ~~$c \geq 10$~~ $10n^2 \leq cn^3$ where $c \geq 1$

~~$n^3 \leq 10n^2$~~ ~~therefore~~ therefore it is $O(n^3)$

~~it is~~ ~~$O(n^3)$~~

d) $\sum_{i=1}^{n} i^k$ is $O(n^{k+1})$ and $\Omega(n^{k+1})$ for k int

$T(n) \leq cn^{k+1}$ where $c \geq 1$

when $i = 8$

$1^k + 2^k + 3^k + 4^k + 5^k + 6^k + 7^k + 8^k \leq c\, 8^{k+1}$

~~therefore~~

therefore it is $O(n^{k+1})$

when $i = 8$

$1^k + 2^k + 3^k + 4^k + 5^k + 6^k + 7^k + 8^k \geq c(8^{k+1})$ where $c \geq 1$

therfore it is $\Omega(n^{k+1})$

e). $P(x)$ $k^{th}$ degree polynomial $c \geq 0$

$P(x) = ax^3 + bx^2 + cx + d$

$P(n) = an^3 + bn^2 + cn + d$

$P(n) \leq cn^3$ for some value of $c$

$an^3 + bn^2 + c_1 n + d \leq cn^3$

therefore it is $O(n^k) \Rightarrow O(n^3)$

for same reason

$an^3 + bn^2 + c_1 n + d \leq cn^3$

it is $\Omega(n^k) \Rightarrow \Omega(n^3)$

1.16

a) $n$ $\cancel{< \left(\frac{3}{2}\right)^n} < 17$ $< \sqrt{n}\log^2 n <$

b) $\sqrt{n}$ $\quad \frac{n}{\log n} < \log^2 n < \left(\frac{1}{3}\right)^n < 17 < \log\log(n) < \log n < n^{1/2} < n < \left(\frac{3}{2}\right)^n$

c) $\log n$

d) $\log\log(n)$

e) $\log^2 n$ $\quad \frac{n}{\log n} < \log^2 n < \left(\frac{1}{3}\right)^n < 17 < \log\log n < \log n < n^{1/2} < \sqrt{n}\log^2 n < \left(\frac{3}{2}\right)^n$

f) $\frac{n}{\log n}$

g) $\sqrt{n}\log^2 n$

h) $\left(\frac{1}{3}\right)^n$

i) $\left(\frac{3}{2}\right)^n$

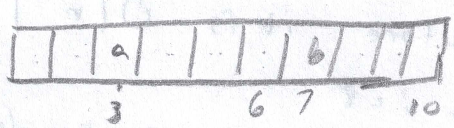
j) 17

1.18) max(i, n)

returns largest element in position $i$ through $i+n-1$ of an int array A

$n$ is a power of 2

max(3, 8)



3, 3+8-1

(3, 4)

m1 = max(3, 4) => (3, 6) => max(3, 2) => max(3, 1)

m2 = max( $\underbrace{3+4}_{7}$, 4) 7, 10 => max(7, 2) => max(7, 1)

$T(n)$, $T(1)$

$T(n) = \max(i, n)$

$T(1) = c$ e index at $A[i]$ some constant

$T(2) = T(n/2) + T(n/2) + b$

$= 2T(n/2) + b$

$T(n) = 2T(n/2) + b$

$T(n) = nc + \frac{n}{2}b \geq kn$

$\Omega(n)$

$T(n) \quad nc + \frac{n}{2}b \leq kn$

$O(n)$

$T(1) = c$

$T(2) = 2T(1) + b$

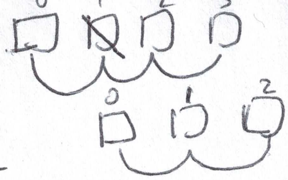
$= 2c + b$

$T(4) = 4c + 2b$

$T(8) = 8c + 4b$

is $O(n)$ and $\Omega(n)$

2.9)



while p <> end (L)

if ~~element~~ L [p] = x

delete (p, L)

p : next (p, L)

the following ~~procedure~~ procedure skips the next node if it founds a match. for example if position p and p+1 has x it will only remove x from position p

to fix this

if RETRIEVE (p, L) = x then

DELETE (p, L)

else

p := NEXT (p, L)

2.11) L is a List

p, q, r positions

n → length of the list

p := first (L) ⇒ 1 , 1

while p <> END (L) do begin

q := p

while q <> END (L) do begin

q := next (q, L) 1

r := first (L) 1

while r <> q do

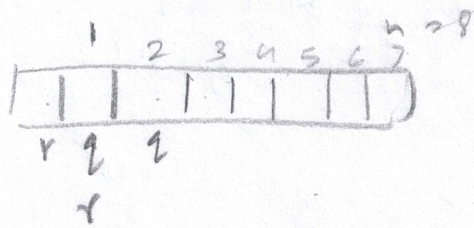
r := next (r, L) → 7

end

end

p := Next (p, L)

end



END will execute = 2n

~~first and Next will~~

first will = $n^2$

next will = $n^3$