Laboratory 4

Arkai Julian Ariza Millares

1. Cormen, Leiserson, Rivest and Stein Exercises

1.1. Exercise 1.2 - 2

Insertion sort requires $8n^2$ steps and merge sort $64n\log_2 n$. This means that insertion sort is faster when $8n^2 < 64n\log_2 n$ holds. This occurs when $n < 8\log_2 n$ or when $n \leq 43$. For this values insertion sort is faster than merge sort.

1.2. Exercise 1.2 - 3

We want that $100n^2 < 2^n$. If $n = 14$ then this expression is $100 * 14^2 = 19600 > 2^{14} = 16384$. If $n = 15$ then $100 * 15^2 = 22500 < 2^{15} = 32768$.So, the answer is $n = 15$.

1.3. Problem 1 - 1

1.3.1. One microsecond

	1 second	1 minute	1 hour	1 day	1 month	1 year	1 century
$\log_2 n$	$2^{1\times10^6}$	$2^{6\times10^7}$	$2^{3{,}6\times10^9}$	$2^{8{,}64\times10^{10}}$	$2^{2{,}592\times10^{12}}$	$2^{3{,}1536\times10^{13}}$	$2^{3{,}15576\times10^{15}}$
$\sqrt{n}$	1×10^{12}	$3{,}6\times10^{15}$	$1{,}29\times10^{19}$	$7{,}46\times10^{21}$	$6{,}72\times10^{24}$	$9{,}95\times10^{26}$	$9{,}96\times10^{30}$
n	1×10^6	6×10^7	$3{,}6\times10^9$	$8{,}64\times10^{10}$	$2{,}59\times10^{12}$	$3{,}15\times10^{13}$	$3{,}16\times10^{15}$
$n\log_2 n$	62746	2801417	133378058	2755157513	71870856404	797633893349	$6{,}86\times10^{13}$
n^2	1000	7745	60000	293938	1609968	5615692	56176151
n^3	100	391	1532	4420	13736	31593	146679
2^n	19	25	31	36	41	44	51
$n!$	9	11	12	13	15	16	17

1.3.2. One nanosecond

	1 second	1 minute	1 hour	1 day	1 month	1 year	1 century
$\log_2 n$	$2^{1\times 10^9}$	$2^{60\times 10^9}$	$2^{3600\times 10^9}$	$2^{86400\times 10^9}$	$2^{2592000\times 10^9}$	$2^{31104000\times 10^9}$	$2^{3110400000\times 10^9}$
$\sqrt{n}$	1×10^{18}	6×10^{19}	$3{,}6\times 10^{21}$	8064×10^{22}	$2{,}592\times 10^{24}$	$3{,}1104\times 10^{25}$	$3{,}1104\times 10^{27}$
n	1000000000	60000000000	$3{,}6\times 10^{12}$	$8{,}64\times 10^{13}$	$2{,}59\times 10^{15}$	$3{,}11\times 10^{16}$	$3{,}11\times 10^{18}$
$n\log_2 n$	39600000	2376000000	$1{,}425\times 10^{11}$	$3{,}42\times 10^{12}$	$1{,}02\times 10^{14}$	$1{,}231\times 10^{15}$	$1{,}231\times 10^{17}$
n^2	31622	1897366	113841995	2732207898	81966236952	$9{,}83\times 10^{11}$	$9{,}83\times 10^{13}$
n^3	3333	200000	12000000	288000000	8640000000	$1{,}03\times 10^{11}$	$1{,}03\times 10^{13}$
2^n	29	1737	104256	2502144	75064320	900771840	90077184000
$n!$	12	713	42804	1027296	30818880	369826560	36982656000

1.4. Problem 3 - 1

a. If we pick any $c > 0$, then , the end behavior of $cn^k - p(n)$ is going to infinity, in particular, there is an n_0 so that for every $n \geq n_0$, it is positive, so , we can add $p(n)$ to both sides to get $p(n) < cn^k 0$.

b. If we pick any $c > 0$, then , the end behavior of $p(n) - cn^k$ is going to infinity, in particular, there is an n_0 so that for every $n \geq n_0$, it is positive, so , we can add cn^k to both sides to get $p(n) > cn^k 0$.

c. We have the previous parts that $p(n) = O(n^k$ and $p(n) = \Omega(n^k)$. So , by known theorem, we have that $p(n) = \Theta(n^k)$.

d.

$$\lim_{n\to\infty} \frac{p(n)}{n^k} = \lim_{n\to\infty} \frac{n^d(a_d + o(1))}{n^k} < \lim_{n\to\infty} \frac{2a_d n^d}{n^k} = 2a_d \lim_{n\to\infty} n^{d-k} = 0 \tag{1.1}$$

e.

$$\lim_{n\to\infty} \frac{n^k}{p(n)} = \lim_{n\to\infty} \frac{nk}{n^d O(1)} < \lim_{n\to\infty} \frac{n^k}{n^d} = \lim_{n\to\infty} n^{k-d} = 0 \tag{1.2}$$

2. Dasgupta, Papadimitoru Varizani Exercises

2.1. Exercise 0.1

a. Both use $O(n)$ notation, so it is $f = \Theta(g)$.

b. Power $\frac{1}{2} < \frac{2}{3}$, so $f = O(g)$.

c. Both use $O(n)$, so $f = \Theta(g)$.

d. Polynomial eliminates logarithm, so $f = \Theta(g)$.

e. Both are $O(n \log_2 n)$, so $f = \Theta(g)$.

f. Both are $O(n \log_2 n)$, so $f = \Theta(g)$.

g. f is superior to g, so $f = \Omega(g)$.

h. f is superior to g, so $f = \Omega(g)$.

i. f is superior to g, so $f = \Omega(g)$.

j. f is superior to g, so $f = \Omega(g)$.

k. f is superior to g, so $f = \Omega(g)$.

l. Power $\frac{1}{2}$ dominates $\log_2 n$, so $f = O(g)$.

m. 3^n dominates 2^n, so $f = O(g)$.

n. Both use $O(n)$, so $f = \Theta(g)$.

o. f is superior to g, so $f = \Omega(g)$.

p. $n^{\log_2 \log_2 n} < n^{\log_2 n}$, so $f = O(g)$.

q. Both use $O(n)$, so $f = \Theta(g)$.

2.2. Exercise 0.2

Geometric series of $g(n)$ is $g(n) = \frac{c^{n+1}-1}{c-1}$.

a. If $c < 1$ then

$$\lim_{x\to\infty} g(n) = \frac{0-1}{C-1} = \frac{-1}{C-1} \tag{2.1}$$

so, by the rule of aymptotic notation $g(n) = \Theta(1)$.

b. If $c = 1$ then

$$\lim_{x\to\infty} g(n) = \frac{c^{n+1}-1}{C-1} = n+1 \tag{2.2}$$

so, $g(n) = \Theta(n)$.

c. If $c > 1$ then

$$\lim_{x\to\infty} g(n) = \frac{(\frac{1}{c^n})(c^{n+1}-1)}{C-1} = \frac{(c^{n+1}-1)}{c^{n+1}-c^n} = \frac{c}{c-1} \tag{2.3}$$

so, $g(n) = \Theta(c^n)$.

3. Demostration

3.1. Recursive sustitution method

3.1.1. T(0) = 0

$T(n) = 2T(n-2)+2 = 2(2T(n-4)+2) = 4T(n-4)+6 = 4(2T(n-6)+2)+6 = 8T(n-6) + 14 = 8(2T(n-8) + 2) + 14 = 16T(n-8) + 30 = 2^k - 2$

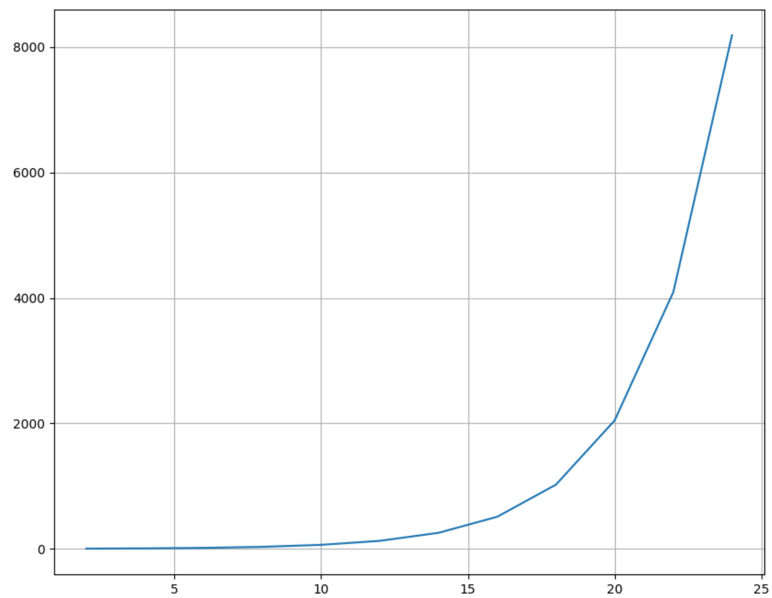
3.1.2. T(0)=1

$T(n = 2T(n-2)+2 = 2(2T(n-4)+2 = 4T(n-4)+6 = 4(2T(n-6)+2)+6 = 8T(n-6) + 14 = 8(2T(n-8) + 2) + 14 = 16T(n-8) + 30 = {}''^{k+1} - 2$

3.2. Graphical method

```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt

g = plt.figure(figsize=(10, 8))
ax = g.add_subplot(111)

def graph(n):
  if(n==0):
    return 0
  else:
    return 2*graph(n-2)+2

t = np.arange(2, 26, 2)
t2 = []
for i in range(0,12):
  t2.append(graph(t[i]))

plt.plot(t,t2)
plt.grid(True)
fig.savefig('graph.png')
```

Figura 1: Result : 2^K - 2




```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt

g = plt.figure(figsize=(10, 8))
ax = g.add_subplot(111)

def graph(n):
  if(n==0):
    return 1
  else:
    return 2*graph(n-2)+2

t = np.arange(2, 26, 2)
t2 = []
for i in range(0,12):
  t2.append(graph(t[i]))
```

```
plt.plot(t,t2)
plt.grid(True)
fig.savefig('graph.png')
```

Figura 2: Result : 2^{K+1} - 2

(1).png