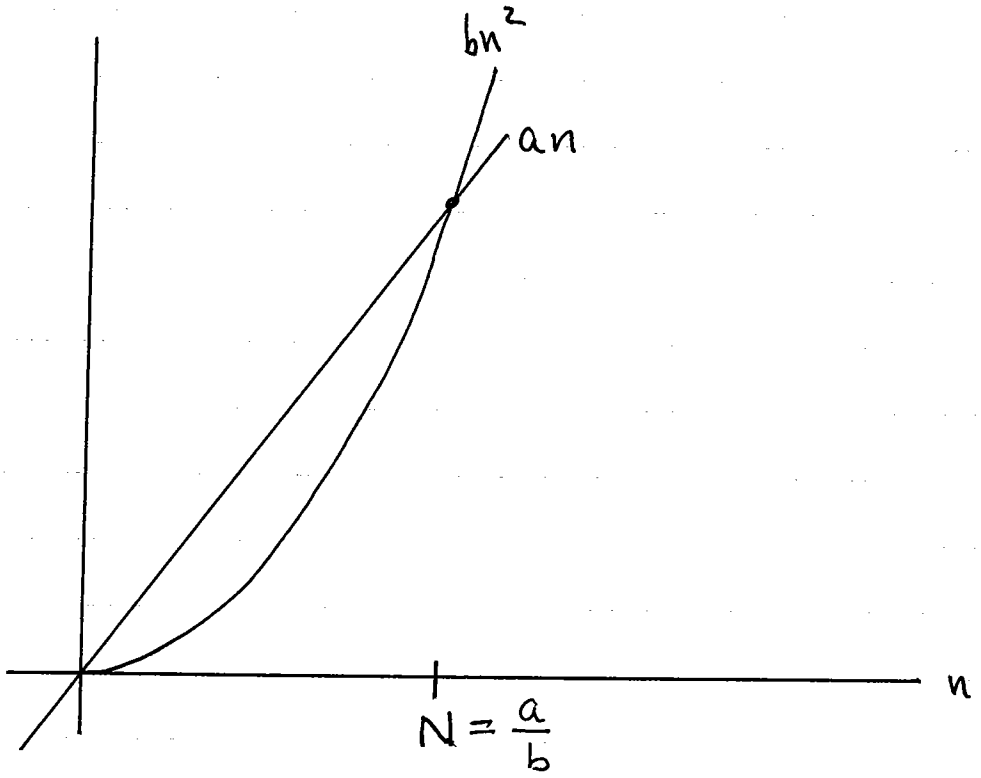
WHATEVER a AND b ARE, THERE EXISTS AN INTEGER N SUCH THAT FOR ALL $n > N$:

$$bn^2 > an$$

EX COMPARE $10n$ TO $(.1)n^2$

n	$10n$	$(.1)n^2$
1	10	.1
10	100	10
$N = 10^2$	10^3	10^3
10^3	10^4	10^5
10^4	10^5	10^7
10^5	10^6	10^9
⋮		



MORE GENERALLY, ANY PARABOLA IS EVENTUALLY ABOVE ANY LINE.

THEOREM

GIVEN FUNCTIONS $an+b$ AND cn^2+dn+e WHERE $a>0$, $c>0$, THERE EXISTS AN $N>0$ SUCH THAT FOR ALL $n>N$:

$$cn^2+dn+e > an+b$$

THE FUNCTION cn^2+dn+e IS SAID TO BE OF <u>ORDER n^2</u>, WHILE $an+b$ IS OF <u>ORDER n</u>. THE SET OF ALL FUNCTIONS WHICH ARE OF ORDER n^2 IS DENOTED $\Theta(n^2)$, WHILE THE SET OF FUNCTIONS OF ORDER n IS $\Theta(n)$.

<u>DEFINITION:</u>

A FUNCTION $f(n)$ IS SAID TO BE OF <u>ORDER $g(n)$</u> IF THERE ARE POSITIVE CONSTANTS c_1, c_2, AND N SUCH THAT FOR ALL $n>N$:

$$0 \le c_1 g(n) \le f(n) \le c_2 g(n)$$

WE OFTEN DENOTE THIS BY WRITING $f(n) = \Theta(g(n))$.

EX. $2n^2 - 3n - 4 = \Theta(n^2)$ SINCE FOR ALL $n > 4$:

$$0 \leq 1 \cdot n^2 \leq (2n^2 - 3n - 4) \leq 2n^2 .$$

IN GENERAL IF $P(n)$ is ANY POLYNOMIAL OF DEGREE k,

$$P(n) = a_k n^k + a_{k-1} n^{k-1} + \cdots + a_1 \cdot n + a_0$$

THEN $P(n)$ is OF ORDER n^k : $P(n) = \Theta(n^k)$.

ANY ALGORITHM WHOSE TIME EFFICIENCY is $\Theta(n)$ is INHERENTLY BETTER THAN ONE WITH TIME EFFICIENCY $\Theta(n^2)$.

TO SEE THIS SUPPOSE WE HAVE A PC WHICH DOES 100 MEGAFLOPS (MILLION FLOATING POINT OPERATIONS PER SECOND), AND A CRAY DOING 500 GIGAFLOPS (BILLION F.P.O.S), I.E. THE CRAY is 5000 TIMES FASTER THAN THE P.C.

PC: $10^8 \frac{OP}{SEC} \rightarrow 10^{-8} \frac{SEC}{OP}$

CRAY: $5 \cdot 10^{11} \frac{OP}{SEC} \rightarrow 2 \cdot 10^{-12} \frac{SEC}{OP}$.

INPUT SIZE	$\Theta(n)$	$\Theta(n^2)$
n	PC	CRAY
100	10^{-6} SEC.	$2 \cdot 10^{-8}$ SEC.
1,000	10^{-5}	$2 \cdot 10^{-6}$
10^4	10^{-4}	$2 \cdot 10^{-4}$
10^5	10^{-3}	$2 \cdot 10^{-2}$
10^6	10^{-2}	2
10^7	10^{-1}	200
10^8	1 SEC.	20 000 SEC = 5.5 HRS
10^9	10 SEC.	550 HRS.

WE HAVE FOUND A MEASURE OF TIME EFFICIENCY WHICH IS INDEPENDENT OF THE COMPUTING DEVICE.

NOW LETS LOOK AT THE DATA CLEAN-UP ALGORITHMS ONCE AGAIN. WE CONSIDER TWO BASIC OPERATIONS: COMPARING TWO NUMERICAL VALUES, AND COPYING A VALUE TO A VARIABLE

ALL THREE ALGORITHMS EXAMINE ALL n ELEMENTS OF THE INPUT LIST, SO ALL THREE DO n COMPARISONS.

SHUFFLE LEFT

BEST CASE: NO ZEROS, NO COPIES
$\therefore n+0 = n$ OPERATIONS

WORST CASE: ALL ZEROS, $n(n-1)$ COPIES
$\therefore n+(n^2-n) = n^2$ OPERATIONS

IT IS DIFFICULT TO DEFINE WHAT IS MEANT BY 'AVERAGE CASE' FOR THIS PROBLEM SINCE THE AMOUNT OF WORK DEPENDS ON BOTH THE NUMBER OF ZEROS AND THEIR LOCATIONS IN THE LIST.

COPY OVER

BEST CASE: ALL ZEROS, NO COPIES
$\therefore n+0 = n$ OPERATIONS

WORST CASE: NO ZEROS, n COPIES
$n+n = 2n$ OPERATIONS

CONVERGING POINTERS

BEST CASE: NO ZEROS, NO COPIES
$n+0 = n$ OPERATIONS

WORST CASE: ALL ZEROS, $n-1$ COPIES
$n+(n-1) = 2n-1$ OPERATIONS

		SHUFFLE LEFT	WAY OVER	CONVERGING POINTERS
TIME	BEST	$\Theta(n)$	$\Theta(n)$	$\Theta(n)$
	WORST	$\Theta(n^2)$	$\Theta(n)$	$\Theta(n)$
	AVERAGE	$\Theta(n^2)$	$\Theta(n)$	$\Theta(n)$
SPACE		n	$2n$	n

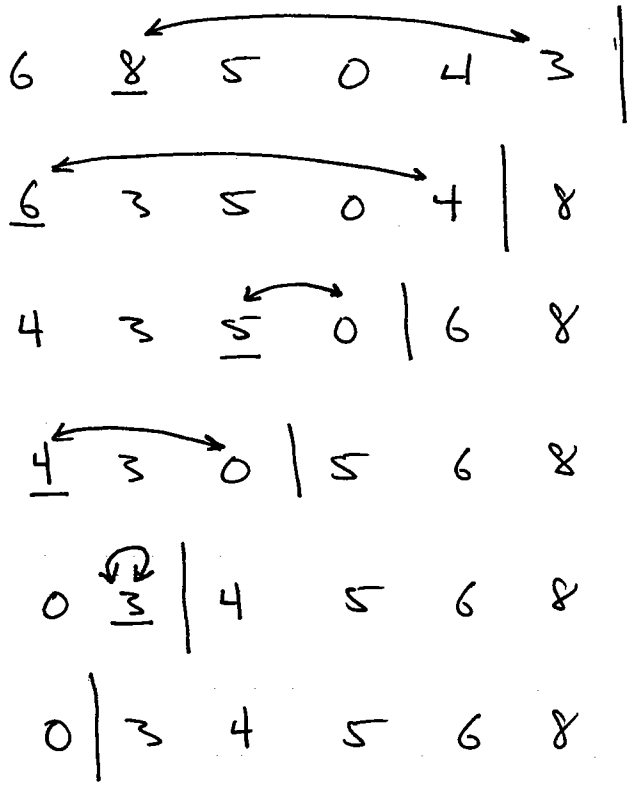
SORTING

THE PROBLEM OF SORTING A LIST OF NUMBERS IS MUCH STUDIED IN COMPUTER SCIENCE. THERE ARE MANY ALGORITHMS WHICH SOLVE THIS PROBLEM. WE WILL SORT IN INCREASING ORDER.

SELECTION SORT

WE DIVIDE THE LIST INTO TWO SECTIONS: SORTED AND UNSORTED, INITIALLY THE SORTED SECTION IS EMPTY. A MARKER FIXES THE BOUNDRY BETWEEN SORTED AND UNSORTED SECTIONS, AND IS TO THE RIGHT INITIALLY. WE REPEATEDLY FIND THE MAXIMUM ELEMENT IN THE UNSORTED SECTION AND EXCHANGE IT WITH THE RIGHTMOST ELEMENT IN THE UNSORTED SECTION, THEN MOVE THE MARKER TO THE LEFT. WE STOP WHEN THERE IS JUST ONE ELEMENT IN THE UNSORTED SECTION.

EX. $n = 6$



	INDEX OF TOP OF UNSORTED SEC
6 8 5 0 4 3 |	6
6 3 5 0 4 | 8	5
4 3 5 0 | 6 8	4
4 3 0 | 5 6 8	3
0 3 | 4 5 6 8	2
0 | 3 4 5 6 8	1

EX. $n = 7$

	TOP
1 0 6 3 2 5 3 |	7
1 0 3 3 2 5 | 6	6
1 0 3 3 2 | 5 6	5
1 0 2 3 | 3 5 6	4
1 0 2 | 3 3 5 6	3
1 0 | 2 3 3 5 6	2
0 | 1 2 3 3 5 6	1

TOP IS THE POSITION OF THE MARKER, AND THE INDEX OF THE NEXT ELEMENT TO BE EXCHANGED WITH THE MAXIMUM IN UNSORTED SECTION.