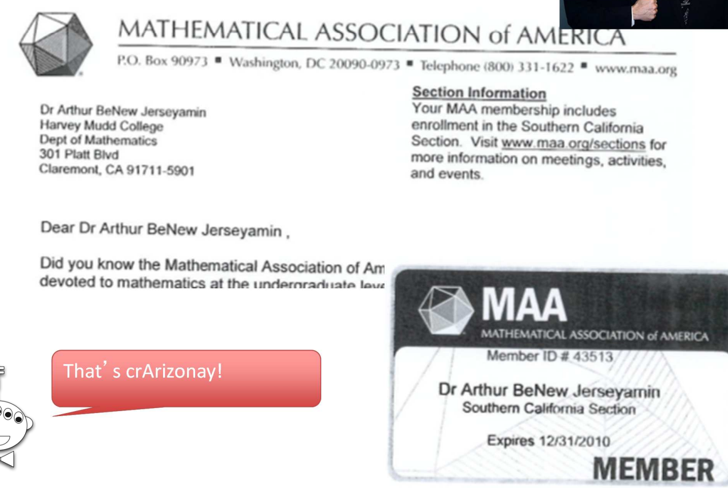
The CS 5 Times

CS 5 Penguin Knocked Into Icy Waters

Claremont (AP): "Our friendly CS 5 penguin was knocked into icy waters by an evil rival," claimed a distraught HMC CS professor. Geoff and Zach are investigating the incident, which was caught on security cameras. Geoff will fly first-class to Antarctica to collect further evidence.

True Story...



MATHEMATICAL ASSOCIATION of AMERICA
P.O. Box 90973 ▪ Washington, DC 20090-0973 ▪ Telephone (800) 331-1622 ▪ www.maa.org

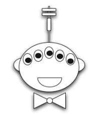
Dr Arthur BeNew Jerseyamin
Harvey Mudd College
Dept of Mathematics
301 Platt Blvd
Claremont, CA 91711-5901

Section Information
Your MAA membership includes enrollment in the Southern California Section. Visit www.maa.org/sections for more information on meetings, activities, and events.

Dear Dr Arthur BeNew Jerseyamin ,

Did you know the Mathematical Association of Am
devoted to mathematics at the undergraduate leve

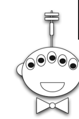
MAA
MATHEMATICAL ASSOCIATION of AMERICA
Member ID # 43513
Dr Arthur BeNew Jerseyamin
Southern California Section
Expires 12/31/2010
MEMBER



That's crArizonay!

```
Date: Wed, 10 Feb 2010 10:39:45 -0800 (PST)
From: Arthur Benjamin <benjamin@math.hmc.edu>
To: Tina Straley <tstraley@maa.org>
Cc: Arthur Benjamin <benjamin@math.hmc.edu>
Subject: amusing (?) typo


Dear Tina (or should that be TIndianaa?):

Yesterday I received the attached letter from MAA membership with my
"new" name, "Arthur BeNew Jerseyamin". Apparently the NJ in the middle
of my last name was replaced by New Jersey.

I laughed it off, but I worry that this could be a bigger problem, and
thought you should know.

Art (or should that be Arkansast?)
```

Sorting Algorithms!



This is my *sort* of algorithm!

Insertion Sort:

`isort([42, 10, 1, 6, 5])`

Through the wonders of recursion!

`[1, 5, 6, 10]`

`insert(42, [1, 5, 6, 10])`
`[1, 5, 6, 10, 42]` A new sorted list!

Write the program (both functions) on your worksheet!

Solution follows…

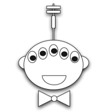
Analyzing Algorithms!

```
def member(item, List):
    if List == []:
        return False
    elif List[0] == item:
        return True
    else:
        return member(item, List[1:])
```

What is the ***worst-case*** running time as a function of the length of the input (denoted “n”)?

It’s approximately…

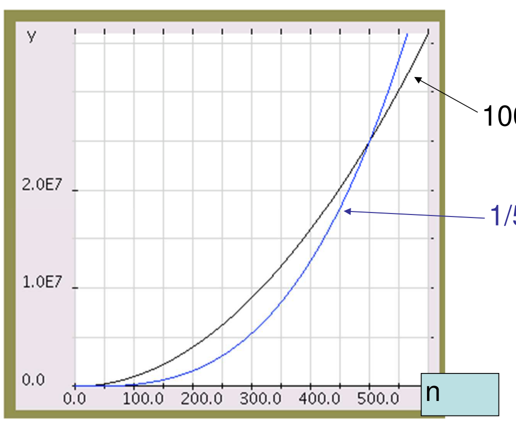
Asymptotic Analysis

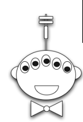


“Asymptotic” is a fancy word!

Expressions	
$3n + 42$ $42n+4242$ $100\ n$	“asymptotically linear”
$(0.1)n^2 + n + 1$ $100\ n^2$	“asymptotically quadratic”
$n^3 - 100n^2 + 2n + 42$ $2n^3 + 10$ $1/5\ n^3$	“asymptotically cubic”

Asymptotic Analysis





Those look like two awesome water-slides!

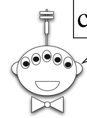
Asymptotic Analysis

Simple (and not-quite-correct) rules:

1. Replace all additive and multiplicative constants by 1
2. Replace constant bases of exponents/logs by 2
3. Discard all but the highest power
    - 2^n beats n^k for any constant k
    - n^k beats n^j for $k > j$
    - n^1 beats log n
    - log n beats 1

Analyzing Algorithms!

Insertion Sort:



We're looking for the worst-case analysis!

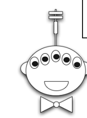
```
isort([42, 10, 1, 65, 5])
```

"the magic of recursion!"

```
[1, 5, 10, 65]
```

Analyzing Algorithms!

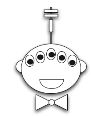
Insertion Sort:



This is amazingly cool stuff!

```
isort([42, 10, 1, 6, 5])
```

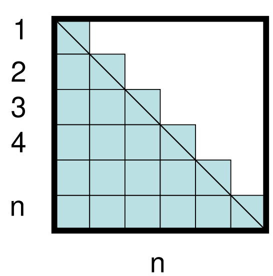
```
insert(42, [1, 5, 10, 65])
[1, 5, 10, 42, 65]
```
A new list!



Let's solve this on the board!

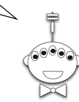
This Space Property of CS 5 Black

1 + 2 + 3 + 4 + … + (n-1) + n = ?



The Alien's Life Advice

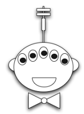
Knowing an obscure fact isn't proof of intelligence



…or even wisdom!

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```



Assume—just for a moment—that the length, n, is a power of 2.

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```

```
msort([42, 3, 1, 5])  msort([27, 8, 2, 7])
```

"the magic of recursion!"

```
[1, 3, 5, 42]     [2, 7, 8, 27]
```

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```

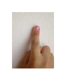
```
msort([42, 3, 1, 5])  msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])
```

```
[
```

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```

```
msort([42, 3, 1, 5])  msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])
```



```
[1,
```

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```

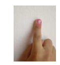
```
msort([42, 3, 1, 5]) msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])
```

```
[1,2
```

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```

```
msort([42, 3, 1, 5]) msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])
```

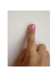


```
[1,2,3
```

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```

```
msort([42, 3, 1, 5]) msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])
```

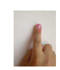


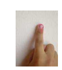
```
[1,2,3,5
```

Mergesort

```
msort([42, 3, 1, 5, 27, 8, 2, 7])
```

```
msort([42, 3, 1, 5]) msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])
```





```
[1,2,3,5,7
```

Mergesort

msort([42, 3, 1, 5, 27, 8, 2, 7])

msort([42, 3, 1, 5]) msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])

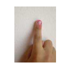
[1,2,3,5,7,8

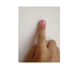
Mergesort

msort([42, 3, 1, 5, 27, 8, 2, 7])

msort([42, 3, 1, 5]) msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])





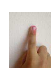
[1,2,3,5,7,8,27

Mergesort

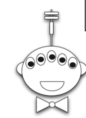
msort([42, 3, 1, 5, 27, 8, 2, 7])

msort([42, 3, 1, 5]) msort([27, 8, 2, 7])

merge([1, 3, 5, 42], [2, 7, 8, 27])



[1,2,3,5,7,8,27,42] ***Done!***



Let's try it out - and let's not even make n a power of 2!

msort([42, 3, 1, 6, 5, 2, 7])

msort([42, 3, 1]) msort([6, 5, 2, 7])

[1, 2, 3, 5, 6, 7, 42]

msort([42, 3, 1, 6, 5, 2, 7])

[1,3,42]

msort([42, 3, 1])

[42]

msort([42])

[1,3]

msort([3, 1])

[3]

msort([3])

[1]

msort([1])

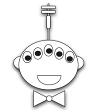
[2, 5, 6, 7]

msort([6, 5, 2, 7])

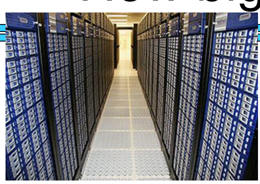
msort([6, 5])

msort([2, 7])

How “Efficient” Is Mergesort?



How big a deal is this?



Geoff’s Super-O-Matic Supercomputer:
100 billion steps/second

	n^2 algorithm	$n \log_2 n$ algorithm
$n = 10^8$	11.5+ **days**	