CS344: Design And Analysis of Computer Algorithms

S. Muthu Muthukrishnan
**Details**

- **Meeting:** Tues, Fri. 11.30 – 12.50PM. SEC118
- **Professor:** Muthu Muthukrishnan, 4452379, Core319.
- **Officehours:** Tuesdays 3.30 – 4.30PM.
- **Contact:**
  - muthu at cs.rutgers.edu
  - http://www.cs.rutgers.edu/~muthu
- **TAs:**
MoreDetails

• **Book:**
  – Cormen, Leiserson, Rivest, Stein.
  http://theory.lcs.mit.edu/~clr/

• **Grading:**
  – 2 Midterms 25% each: March 4, April 8.
  – 1 Final Exam 40%: May 6.
  – Homeworks 10%.
    • Team of X people each.
    • Assignment given on Fridays, due the following Friday.

• **Policy:** Writing “I don’t know” gives you 25% of the points.

• Khachiyan is an excellent lecturer! Both sections are independent.
Syllabus

- Asymptotic analysis of algorithms
- Recursion, induction, recurrence equations.
- Sorting and Selection.
- Hashing and balanced search structures
- Graphs and Graph algorithms
- Dynamic Programming
- String matching.
- Polynomials, matrices.
- NP Completeness
- Assorted topics.
BigPictureofComputerScience

• Theory
  – What is computable? Turing machines, recursive functions.
  – What is efficiently computable? Algorithms, Complexity theory.
  – Abstract math, practical methods.

• Systems
  – Databases, OS, Networks, Programming Languages, Web, ..
  – Theory, tools.

• Popular Science
  – CPUMIPS, GUIs, Blogs, SPAM, ..
• **Design** of efficient algorithms for problems.

• Mathematically **analyze** running time and space used.

• **Lowerbounds**: What is the best possible algorithm, what is “best”?

• Are there resolvable problems that **haveno** efficient algorithms?
Example

- Problem: Maximum subarray sum
- Output: $[i,j]$ such that $A[i] + \ldots + A[j]$ is largest.
Naïve Strategy

- For all pairs $1 \leq i \leq j \leq n$, compute
  - $\sum(i,j) = A[i] + \ldots + A[j]$

- Simple tripleloop will do.

- Running time:
  - Roughly $n^3$. 

```plaintext
For i = 1 to n
    For j = i + 1 to n
        Sum[i,j] = 0;
        For k = i to j
            Sum[i,j] += A[k].
```
**LessNaiveStrategy**

- When you have $\sum(i,j) = A[i] + \ldots + A[j]$, it is easy to get $\sum(i,j+1) = A[i] + \ldots + A[j+1]$.

- **Double loop** will do.

- Running time: $n + (n - 1) + (n - 2) + \ldots + 1$, roughly $n^2$.

What is the fastest running time possible?