Analysis of Algorithms
Fall 2017
Course Information

Amotz Bar-Noy
Department of Computer and Information Science
Brooklyn College

August 29, 2017
Instructor — Amotz Bar-Noy

- **E-mail:** amotz@sci.brooklyn.cuny.edu.
- **Internet:** [http://www.sci.brooklyn.cuny.edu/~amotz/bc-algorithms.html](http://www.sci.brooklyn.cuny.edu/~amotz/bc-algorithms.html)

**Office Hours:** Tuesday 3:00pm–4:00pm, Room 2112a.

**Class Hours:** Tuesday 6:05pm–8:10pm, Room 236NE.
Prerequisite Courses and Knowledge

- A course in data structure
  - Computer and Information Science 6006X [622X]
- A course in discrete structures.
  - 6004X [611X]
**Textbooks**

**Main Textbook**
- 2nd edition and even 1st edition are also good.

---

Amotz Bar-Noy (Brooklyn College)
Textbooks

Main Textbook

  - 2nd edition and even 1st edition are also good.

Other Books

- “Algorithm Design,” Kleinberg and Tardos, Addison Wesley.
- “Algorithm Design,” Goodrich and Tamassia, Wiley.
- “Introduction to Algorithms a Creative Approach,” Manber, Addison-Wesley.
Online Resources

- **Video Lectures for the main text book from MIT:**
  

- **YouTube Lectures for another course on algorithms from Stanford:**
  
  https://www.youtube.com/playlist?list=PLXFMmlk03Dt7Q0xr1PIAriY5623cKiH7V

- **Problems on Algorithms book by I. Parberry and W. Gasarch:**
  
  http://larc.unt.edu/ian/books/free/poa.pdf

- **Video lectures for Mathematics for Computer Science from MIT:**
  
  https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2010/video-lectures/
Tentative Syllabus

- Introduction; Mathematical Background; Analysis of Algorithms.
- Searching; Order Statistics; Sorting.
- Divide & Conquer; Greedy Algorithms; Dynamic Programming.
- Graphs; Graph Traversals; Minimum Spanning Trees.
- NP-Completeness.
<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>08/29/2017</td>
<td>Mathematical Background</td>
</tr>
<tr>
<td>09/05/2017</td>
<td>Analysis of Algorithms</td>
</tr>
<tr>
<td>09/12/2017</td>
<td>Order Statistics</td>
</tr>
<tr>
<td>09/26/2017</td>
<td>Order Statistics/Sorting</td>
</tr>
<tr>
<td>10/03/2017</td>
<td>Sorting</td>
</tr>
<tr>
<td>10/10/2017</td>
<td>Midterm Exam</td>
</tr>
<tr>
<td>10/17/2017</td>
<td>Divide and Conquer</td>
</tr>
<tr>
<td>10/24/2017</td>
<td>Greedy Algorithms</td>
</tr>
<tr>
<td>10/31/2017</td>
<td>Dynamic Programming</td>
</tr>
<tr>
<td>11/07/2017</td>
<td>Graphs</td>
</tr>
<tr>
<td>11/14/2017</td>
<td>Graph Traversals</td>
</tr>
<tr>
<td>11/28/2017</td>
<td>Minimum Spanning Trees</td>
</tr>
<tr>
<td>12/05/2017</td>
<td>NP-Completeness</td>
</tr>
<tr>
<td>12/12/2017</td>
<td>To Be Determined</td>
</tr>
<tr>
<td>12/19/2017</td>
<td>Final Exam</td>
</tr>
</tbody>
</table>
Grading

Percentages

This is only a **guideline**, percentages and rules may change during the semester as needed.

The final grade will be composed of the following 5 components:

- ≈ 40% – 60% final exam.
- ≈ 20% – 30% mid-term exam.
- ≈ 0% – 20% quizzes.
- ≈ 10% – 10% assignments.
- ≈ 0% – 20% programming project.

General Formula

\[
\text{final\%} = 100 - \text{midterm\%} - \text{assignments\%} - \text{quizzes\%} - \text{project\%}.
\]

Final exam grade dominates: only grades that are greater than the final exam grade count!
Grading

Percentages

- This is only a **guideline**, percentages and rules may change during the semester as needed.

- The final grade will be composed of the following 5 components:
  - \( \approx 40\% - 60\% \) final exam.
  - \( \approx 20\% - 30\% \) mid-term exam.
  - \( \approx 0\% - 20\% \) quizzes.
  - \( \approx 10\% - 10\% \) assignments.
  - \( \approx 0\% - 20\% \) programming project.

General Formula

- \( \text{final\%} = 100 - \text{midterm\%} - \text{assignments\%} - \text{quizzes\%} - \text{project\%}. \)

- Final exam grade dominates: only grades that are greater than the final exam grade count!
There could be two types of quizzes:

- At the beginning of the class to check what you learned in the previous week.
- At the end of the class to check what you learned during the class.

There might be no announcements regarding quizzes.

The number of quizzes has not yet been determined.
Answer a question in an exam, in a quiz, or in an assignment:
- Only within the given space for the answer.
- Using a readable text with normal size font.
- You get 20% of the value if you leave the answer blank.
- You get no points for a wrong answer.
Preparing Assignments

- Type the answers or use a *readable* hand writing.
- Do the assignments alone if you can.
- Get help if necessary.
- You **must** understand everything you write.
Reading and Practicing Assignment

- Refresh your algorithmic knowledge and mathematical foundations.
  - In the second edition read Chapters 1–4 (without 4.4) and Appendices A–D (without C.5). In the first edition read Chapters 1–5 (without 4.4).

- Practice by solving some or all of the problems in the books and online resources.

Refresh your algorithmic knowledge and mathematical foundations.

  - In the second edition read Chapters 1–4 (without 4.4) and Appendices A–D (without C.5). In the first edition read Chapters 1–5 (without 4.4).

Practice by solving some or all of the problems in the books and online resources.

- Solve problems in Chapters 1–5 of the online book “Problems on Algorithms,” by Ian Parberry.
  
  http://larc.unt.edu/ian/books/free/poa.pdf
**Reading and Practicing Assignment**

- Refresh your algorithmic knowledge and mathematical foundations.
  - In the second edition read Chapters 1–4 (without 4.4) and Appendices A–D (without C.5). In the first edition read Chapters 1–5 (without 4.4).

- Practice by solving some or all of the problems in the books and online resources.
  - Solve problems in Chapters 1–5 of the online book “Problems on Algorithms,” by Ian Parberry.
    
    [http://larc.unt.edu/ian/books/free/poa.pdf](http://larc.unt.edu/ian/books/free/poa.pdf)

- Watch online videos about “Mathematics for Computer Science.”