Institut for Matematik og Datalogi | December 15, 2010
Syddansk Universitet | JFB

Introduction to Computer Science
E10 – Lecture 17

Lecture, November 25, 14:15–16, U71

Lene Monrad Favrholdt lectured on on-line algorithms.

Lecture, November 29, 8:15–10, U37

We will cover the first five sections of chapter 7 in the textbook.

Lecture, December 6, 8:15–10, U37

We will cover section 7.6 in the textbook and begin on chapter 11.

Discussion section: December 15, 12:15-14, in U26. Bring a pocket calculator.

Discuss the following problems (some are from the textbook) in groups of 3 to 4. Actually, only one person in your group needs to bring a calculator. (Think about these problems before coming to discussion section. Prepare your secret encryptions in advance.)

1. This English message was encrypted using a Caesar cipher. Decrypt it.

   YMNX HWDUYTLWFR NX JFXD YT IJHNUMJW.

   Discuss which techniques you used.

2. Each person in the group should choose a secret key for performing encryption with the Caesar cipher (the alphabet is shifted by the amount specified by the key) and choose a secret message (not more than 15 letters). Encrypt your message with the secret key. Give everyone in your group a copy of the encrypted message and let them try to break it. Let them know if the message (and alphabet) is English or Danish.

3. This was entitled "Cold Country". It was encrypted using a monoalphabetic substiution cipher. In such a cipher, the key is permutation of the alphabet, so that you decide according to this key what letter "A" maps to, what letter "B" maps to, etc.

   TOWWJPHJC ZY RXW PHOTWYR ZYPHJC ZJ RXW SFOPC.
   UFYR FB ZR ZY QFIWOWC SZRX ZQW RXFMYHJCY FB
   BWWR CWWD.

   Discuss which techniques you used.

4. A monoalphabetic substitution ciphers works similarly to a Caesar cipher. However, instead of just shifting the alphabet a fixed amount to get the mapping defined for each letter, the alphabet is permuted randomly (reordered). The key says which letter maps to which. If the alphabet has 29 letters, the number of keys is now 29!.

   Each person in the group should choose a secret key for performing encryption with a monoalphabetic substitution cipher and choose a secret message (with between 60 and 80 letters). Encrypt your message with the secret key. Give everyone in your group a copy of the encrypted message and let them try to break it. Let them know if the message (and alphabet) is English or Danish.

5. Do problem 48 on page 611. Try decrypting. What is the problem here if 110 is interpreted in decimal instead of binary?

6. Do problem 49 on page 611. (Note that 111 is an encryption of a message.)

7. Discuss problems 1, 2, 6, and 7 on pages 611–612.

Assignment due 8:15, December 10

Late assignments will not be accepted. Working together is not allowed. (You may write this either in English or Danish.) Submit a single PDF file through the Blackboard system and include your name on the first page. If you submit this assignment more than once, use the same identification number both times. Remember to explain your answers.

As explained in class, in the Minimum Makespan Problem for identical machines, there are m identical machines available. The online algorithm is given a sequence of requests, which consists of jobs, each of which has a given load. The online algorithm must assign jobs to machines in the order they arrive in that sequence. The *makespan* is the total load on the most heavily loaded machine at the end.

The List Scheduling algorithm (LS) places each new job on the machine with lowest load (currently). Thus, with four machines, and the sequence

$$1, 4, 5, 3, 4, 7,$$

LS will put these six jobs so they end up on the following machines:

$$1, 2, 3, 4, 1, 4.$$

Thus, the machine 4 will have load 10. This is the maximum, so the makespan for this sequence is 10. However, the optimal off-line algorithm could put the first two items on the first machine, the third on the second machine, the fourth and fifth on the third machine, and the last on the fourth machine. Then, no machine has load more than 7, so the makespan is 7.

We showed in class that the algorithm LS has a competitive ratio of $2 - \frac{1}{m}$, where m is the number of machines. (Note the competitive ratio is the worst case ratio, over all possible input sequences, of the value (makespan in the case of the Minimum Makespan Problem) the on-line algorithm achieves on the input sequence to the value the optimal off-line algorithm achieves on the same input sequence.) In order to show that the competitive ratio was this high, we showed that if the algorithm gets many small jobs ($m(m-1)$ of load 1) followed by one large job (a job of load m), LS will pack them so each machine has the same number of the small ones, so the last job will be placed on some machine that already has a large load.

Consider the following algorithm, *FirstGreedy*: When a job, J, of load w arrives and the most heavily loaded machine currently has total load s, place the job J as follows:

- If J can be placed on some machine, such that its total load after placing J there is at most s, place J on the first such machine where this is possible.
- Otherwise, place J on the least loaded machine (as with the List Scheduling algorithm (LS)).

1. Show how *FirstGreedy* would place the following sequence of jobs loads on 5 machines, M_1, M_2, M_3, M_4, M_5:

$$2, 1, 3, 1, 5, 2, 5, 4, 2$$

   How much load is on each machine? What is the makespan?

2. Show how LS places that sequence from the previous problem. What is the makespan?
3. Show that *FirstGreedy* has competitive ratio at least $2 - \frac{1}{m}$, where m is the number of machines.
4. (Optional) Show that *FirstGreedy* has a competitive ratio of at most $2 - \frac{1}{m}$, where m is the number of machines.