

## Sample second Midterm Exam

### Open Book and Notes

(15) 1. In class we showed that for the TSP problem, if our costs satisfied symmetry (the x-y cost is the same as the y-x cost), and the triangle inequality (the x-y cost + the y-z cost is always at least the x-z cost), then we could approximate the optimal solution with an error of at most 50%.

a. Give an example graph with costs which do not satisfy these properties where the cost of our approximation scheme (using a Minimum Spanning Tree, matching and euler tour), would yield a tour much greater than twice the optimal cost.

b. Is it likely that we could find an algorithm which would give factor of 2 approximations for TSP for arbitrary costs? Justify your answer.

(30) 2. Suppose that you have been hired to help out the registrar in scheduling classes. Your input is a list of 2,000 classes to be scheduled for the quarter in the form  $c_i, i = 1, 2, \dots, 2,000$  which is the number of students in the class. Each class is a 3 hour per week class which can be scheduled in a MWF or TTh time slot, and each classroom can be scheduled between 8AM and 6PM MWF and 7:30AM to 6PM TTh. There are 200 classrooms and classroom  $i$  can seat  $r_i$  students for  $i = 1, 2, \dots, 200$ .

a) Your goal is to assign each class to a classroom-time slot pair (e.g. ECS122A to 217 ART MWF 4-5) such that if a class of size  $c_i$  is assigned to a room a room of capacity  $r_j$ , then  $r_j \geq c_i$ . Of course each classroom should have at most one class assigned to each time slot.

i) Describe an efficient algorithm to find such an assignment whenever one exists (or tell the registrar that it is impossible to schedule all the classes).

ii) Estimate the time and space needed by your solution.

iii) Given your solution to ii) how much care should be taken in coding this program? Justify your answer.

b) Now suppose that some picky Professors have constraints on when they want to teach (e.g. 122B has to be scheduled T/Th after 10AM). Thus each class now has a legal begin and end time for MWF and T/Th and the class must be scheduled in this time window.

Describe how to modify your solution to a) to solve this new problem or argue that the new problem is NP-hard.

c) Now suppose that the registrar is unhappy with restrictions like those imposed in part b) which make it impossible to schedule all classes. Instead they allow professors to bid for rooms/ time slots (thus for a given class, say 122b, a Professor can offer to pay 100 dollars for a time T/Th after 10AM, if the class is in Physics Geology, and 200 for the same times in Bainer Hall.

The goal now is to find a schedule which schedules all classes (just using the constraints of part a) ), but which also maximizes the total amount the registrar collects from faculty "bribes".

Describe how to modify your solution to a) to solve this new problem or argue that the new problem is NP-hard.

## (20) 3. Vertex cover

Suppose that you have a routine called VC such that  $VC(G,k)$  returns TRUE if the graph  $G$  has a vertex cover of size  $k$  (or smaller) and FALSE otherwise.

a) Suppose you have an  $n$  vertex  $m$  edge graph  $G$ . How many calls to VC are needed to find the SIZE of the smallest vertex cover in  $G$ ? Justify your answer.

b) Suppose that you found  $c$  the SIZE of the minimum vertex cover in  $G$  and you now want to find the actual cover (set of vertices)  $C$  of size  $c$ . Describe how to find a cover of size  $c$  using at most  $n$  calls to the VC routine.

(15) 4. Program Testing/ Verification Suppose that you implemented a version of the FIFO preflow push algorithm to find the maximum flow in a network. We explore making sure your program is correct.

a) Describe three invariants which it would be useful to test as your program is running to make sure the algorithm is working properly. For example, each time you look at a residual arc  $(i, j)$  you could check that  $h[i] \leq h[j] + 1$  (you can't use this as one of your three!).

b) Suppose you wanted to test your algorithm. For a given network how could you efficiently check that you had correctly found the maximum flow? What is the running time of your check on a network with  $n$  nodes and  $m$  edges?

## 5. Strings and space

For the suffix arrays used in chapter 15 assuming a string of length  $n$ , we store the string in an array  $C$  of characters and the suffix pointers in an array  $A$ .

a) how much space does the scheme described by Bentley use (as a function of  $n$ )?

b) If space were tight, how would you modify his scheme to reduce it? What is the new space used and how might this affect performance?

## 6. Primes

When finding duplicate words Bentley needed to find a prime  $> 29,000$ . How would you find such a prime? Justify your choice.