

## Exam 2

*November 11, 2003*

Please write all solutions clearly and legibly in the space provided. The number of points indicates the maximum number of minutes that you should spend on the problem.

For any NP-completeness proof you can only use the following problems for your reduction: CIRCUIT-SAT, FORMULA-SAT, 3-CNF-SAT, CLIQUE, VERTEX-COVER, SUBSET-SUM, PARTITION, HAM-CYCLE, TSP.

1. (30 pts) Let problem “ $L$ ” be the following decision problem. You are given as input an undirected graph  $G = (V, E)$ , a subset  $E' \subseteq E$ , and an integer  $k$ . (*Note:  $G$  can contain self loops and also multiple edges.*) The question is whether or not there is a cycle in  $G$  consisting of at most  $k$  edges that includes each edge in  $E'$ . You are to prove that  $L$  is NP-hard. (So you need not prove that  $L$  is in NP).

More space for #1 (if needed)

2. In this problem we consider the problem MAX-3-SAT in which the input is a 3-CNF-SAT formula  $\phi = C_1 \wedge \cdots \wedge C_r$  over Boolean variables  $x_1, \dots, x_n$ , and an integer  $1 \leq k \leq r$ . The question is whether or not there is an assignment to the boolean variables such that at least  $k$  of the clauses in  $\phi$  are satisfied.
  - (a) (15 pts) Prove that MAX-3-SAT is NP-Complete. (Only one or two sentences should be needed to argue that your reduction is correct.)

(b) (25 pts) Prove the below algorithm is a  $4/3$ -approximation for MAX-3-SAT.

$SAT = \emptyset$  (the set of clauses satisfied so far)  
 $T = \emptyset$  (the literals that are assigned to be true)  
 $Lits = \{x_1, \bar{x}_1, \dots, x_n, \bar{x}_n\}$   
 $Left = \{C_1, \dots, C_r\}$  (clauses not yet satisfied)  
 While some clause in  $Left$  contains a literal in  $Lits$   
     Let  $y$  be the literal in  $Lits$  contained in the most clauses of  $Left$   
     Let  $Satisfied$  be the clauses in  $Left$  that contain  $y$   
      $SAT = SAT \cup Satisfied$   
      $Left = Left - Satisfied$   
      $T = T \cup \{y\}$  (so if  $y = \bar{x}_i$  this sets  $x_i$  to false)  
      $Lits = Lits - \{y, \bar{y}\}$   
 Return any assignment  $A$  in which all literals in  $T$  are true

We define a clause to be *wounded* if it contains  $\bar{y}$  and  $y$  is set to true (i.e.  $y$  is placed in  $T$ ). Note that when a clause  $C_j$  is wounded by placing  $y$  in  $T$ , if instead you had placed  $\bar{y}$  in  $T$  that  $C_j$  would have been satisfied.

- During each step of the while loop what can you say about the number of wounds inflicted as compared to  $|Satisfied|$ ? Then use this to relate  $|SAT|$  to the total number of wounds inflicted.
  
- When the algorithm stops how many wounds have been inflicted? Observe that no clauses in  $Left$  contain any variable not yet assigned a value.
  
- What is the relationship between  $r$ ,  $|Left|$ , and  $|SAT|$ ?

Use the above facts to finish the proof that the given algorithm is a  $4/3$ -approximation. Let  $C$  be the number of clauses satisfied by  $A$  and let  $C_*$  be the number of clauses satisfied by an optimal solution.

THIS PROBLEM IS REQUIRED ONLY FOR CS 539 STUDENTS. (If a CS 441T student does the next problem we will note it as extra credit, but no points will be added to your total.)

3. (5 pts) Here we continue with problem 2b. Suppose that  $\phi = C_1 \wedge \cdots \wedge C_r$  but now each clause is a conjunction of at least  $k$  literals. Derive the best approximation ratio you can for the algorithm given in 2b for this situation. (Explain any claims and show your work but no complete proofs needed.)

Problem	Points Possible	Points Received
1	30	
2a	15	
2b	25	
3	5	
total	70 + 5	