CS172 Computability & Complexity (Spring'09)

Instructor: Mihai Pătraşcu GSI: Omid Etesami

Problem Set 9 *Due: Monday, April 20*

1. Abdi and Hassan, two pirates off the coast of Somalia, have just comandeered another ship. On board, there are n objects they take interest in. Object i is worth D_i dollars, where D_i is a natural number. Since the ship came from a third-world country, the objects are not worth very much, $\sum_{i=1}^{N} D_i \leq 10000$.

Abdi and Hassan must split the bounty such that both earn the exact same dollar value (i.e. the objects Abdi gets have total value equal to the total of what Hassan gets). Objects are indivisible.

Sometimes, it is impossible to split the bounty in two equal parts. Then, the pirates may leave some objects on the ship, and split the rest equally. (This is always possible: in the worst case, they may leave all objects on the ship, and they each get zero dollars.)

Subject to the equality constraint, Abdi and Hassan want to get as many objects as possible from the ship.

Write a program that finds the optimal solution. Your program should run in under 5 seconds when $n = 100$ (so you will probably want to do something polynomial).

The input will look like: N <new-line> D_1 D_2 $\ldots$ <new-line>

The first line of the output should specify the maximum number M of objects that Abdi and Hassan can take from the ship. On the next M lines, write the index of those objects, followed by **A** if the object is given to Abdi, or **H** if the object is given to Hassan. The value given to each must be equal, and M should be the maximum possible under this rule.

Email the source code (in C, Java, or Pyhton) to `mip@alum.mit.edu` in an email with subject `CS172-HW9-P1`. *Also* send a brief description of how your algorithm works.

2. Merlin has a set of n integers, $x_1, \ldots, x_n$, but he doesn't want to tell you the set. Instead, he is willing to reveal the set of differences $|x_i - x_j|$ for all i, j. Notice that this is a set of $\binom{n}{2}$ values, and you do not know which difference comes from which x_i and x_j (the set is just given in sorted order).

Find the original set; if multiple solutions exist, just output one. You probably want to use a backtracking algorithm.

The input will look like: n <new-line> d_1 d_2 $\ldots$ $d_{\binom{n}{2}}$ <new-line>. The differences are in sorted order $d_1 \leq d_2 \leq \ldots$. You output should be one line containing: $x_1 x_2 \ldots x_n$.

Email the source code (in C, Java, or Pyhton) to `mip@alum.mit.edu` in an email with subject `CS172-HW9-P2`. Also send:

(a) a brief description of how your algorithm works.

(b) the worst-case complexity of your algorithm, using O-notation.

(c) what is the largest n such that your algorithm typically finishes in under one second on random instances of size n? To construct a random instance: generate each x_i randomly between 1 and 10^8, then compute all differences and sort them.

Note: With an optimized backtracking, it is possible to run in one second for instances of thousands of values, even if the worst case is prohibitive. (However, any correct solution will get you full credit.)

3. Consider problem 1. without the upper bound on the sum of D_i (i.e. $\sum_{i=1}^{N} D_i$ can be arbitrary). Show that the problem is NP-complete.

4. Let $DOUBLESAT = \{\phi \mid \text{the formula } \phi \text{ has at least two satisfying assignment }\}$. Show that $DOUBLESAT$ is NP-complete.

5. We know that, given an NFA with n states, we can produce an equivalent DFA with 2^n states. However, for some NFAs, there are smaller DFAs that could be produced. Can we compute the smallest DFA equivalent to a given NFA? Consider the following language:

$$X = \{(A, k) \mid A \text{ is an NFA and } (\exists) B \text{ a DFA with} \leq k \text{ states such that } L(A) = L(B)\}$$

(a) Show that X is decidable.

(b) Show that X is NP-hard, i.e. if $X \in$ P, then P=NP.

Hint: Given a CNF formula, show how to construct an NFA that accepts any *non-satisfying* assignment. (If the formula has n variables and m clauses, your NFA will have size around $m \cdot n$.) If the formula is unsatisfiable, what is the smallest equivalent DFA?