CSE 6331 – Algorithms – Spring, 2015 – Prof. Supowit

Homework 7 – Due: Wednesday, March 11

Recall the Set Covering problem:

INPUT: a finite set X, where $n = |X|$, and a family F of subsets of X.

OUTPUT: a minimum size cover of X.

Also, recall our greedy heuristic for it:

$k \leftarrow 0$;
$U \leftarrow X$; /* U is the set of elements of X not yet covered. */
<u>while</u> $U \neq \varnothing$ <u>do</u> <u>begin</u>
select some $T_{i+1} \in F$ that maximizes $|T_{i+1} \cap U|$;
$U \leftarrow U - T_{i+1}$;
$i \leftarrow i + 1$
<u>end</u> ;

<u>return</u>($\{T_1, T_2, \ldots, T_i\}$).

1. Describe a way to implement this greedy heuristic in time $O\left(\sum_{S \in F} |S|\right)$; that's big-$O$.

You'll need to describe your data structures in sufficient detail.

2. Recall that we saw in class an infinite sequence of inputs to this problem for which $OPT = \sqrt{n}$ and $GRE \in \Theta\left(\sqrt{n} \times \log n\right)$.

Describe an infinite sequence of inputs for which $OPT = 2$ and $GRE \in \Theta(\log n)$.

3. Consider the alphabet $\{a_1, a_2, \ldots, a_n\}$ where the frequency of a_i is proportional to the i^{th} Fibonacci number. That is,

$$f[a_i] = \frac{F_i}{\sum_{j=1}^{n} F_j} \qquad \forall 1 \leq i \leq n \ ,$$

where

$$F_1 = F_2 = 1$$

and

$$F_k = F_{k-1} + F_{k-2} \quad \forall k \geq 3 \ .$$

Give an optimal binary prefix code for this alphabet; that is, for each character of the alphabet, give its binary encoding.