SUBSET-SUM

and Related Problems

SUBSET-SUM

- Recall, the subset-sum problem is the following language:

$$\{\langle S,t\rangle : S=\{x_1,\dots,x_n\}, \text{and for some } \{y_1,\dots,y_l\}\subseteq\{x_1,\dots,x_n\}, \sum_{i=1}^{l} y_i = t\}$$

- Inputs: an integer value (target) t, and a set of integers $a_1,\dots,a_n$
- Output: **YES** if there is a subset that adds up to t, **NO** otherwise

SUBSET-SUM is in NP

- Proof: Sipser Theorem 7.25

SUBSET-SUM is in NPc

- Proof: Sipser Theorem 7.56

- We show that $3SAT \leq_p$ *SUBSET-SUM*

- How?

- Given a 3CNF formula $\emptyset$, we want (**within polytime** in the length of $\emptyset$) to **create(build)** an instance of SUBSET-SUM (dependent on $\emptyset$): $\langle S_\emptyset, t_\emptyset \rangle$ such that $\emptyset$ is satisfiable iff $S_\emptyset$ has a subset that adds up to $t_\emptyset$.

From $\emptyset$ to $\langle S_\emptyset, t_\emptyset \rangle$

- Suppose it has variables $x_1, \ldots, x_l$ and clauses $c_1, \ldots, c_k$

- $t_\emptyset = \underbrace{11 \ldots 1}_{l} \underbrace{33 \ldots 3}_{k}$ (decimal number)
- Example: If $l = 1$ and $k = 1$, then $t_\emptyset$ is thirteen

If $\emptyset = (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_3 \vee \neg x_4 \vee x_5)$, then

$$t_\emptyset = 1111133$$

$S_{\emptyset} = ??$

(1) If x_i is a variable in the formula $\emptyset$ and the literal x_i appears in the clauses $c_{j_1}, \ldots, c_{j_r}$, then include in $S_{\emptyset}$ the number $Y_i =:$

$$1\,\underbrace{00\ldots0}_{l-i}\,\underbrace{0\ldots1\ldots010\ldots0}_{k}$$

Where in red, the 1's are in the positions $j_1, \ldots, j_r$

(2) Do the exact same for the literal $\neg x_i$. Call the resulting number Z_i

(3) For each clause c_j, include $D_j = 1\,\underbrace{00\ldots0}_{k-j}$ and $H_j = 2\,\underbrace{00\ldots0}_{k-j}$

This construction works

(1) If Ø is satisfiable, we can find a subset T of $S_{\emptyset}$ such that the sum of the elements in T equals $t_{\emptyset}$

(2) Conversely, if we can find a subset T of $S_{\emptyset}$ such that the sum of the elements in T equals $t_{\emptyset}$, then Ø is satisfiable

(1)

- Suppose that ∅ is satisfiable (by some assignment); here we build *T*
- If x_i is assigned TRUE, include Y_i in *T*
- If x_i is assigned FALSE, include Z_i in *T*
- With the elements included so far, the sum of the elements in *T* should be something like $\underbrace{11 \ldots 1}_{l}\underbrace{23 \ldots 1 \ldots 3}_{k}$

(the last k digits are made from 1,2,3)

- If needed, include in *T* from D_i and H_i what brings up the sum to $\underbrace{11 \ldots 1}_{l}\underbrace{33 \ldots 3}_{k}$

(2)

- Suppose we can find a subset *T* of $S_\emptyset$ such that the sum of the elements in *T* equals $t_\emptyset$

- We build an assignment which satisfies ∅

- If *T* contains Y_i , then assign x_i to TRUE

- If *T* contains Z_i , then assign x_i to FALSE

(2) Clarified

- Note that all we know about T here is that it is a collection of Y_i ’s, Z_j’s, D_m’s, and H_n ’s
- They all add up to $\underbrace{11\ldots1}_{l}\,\underbrace{33\ldots3}_{k}$
- The 1’s at the beginning imply that for each variable x_i , exactly one of Y_i or Z_i is in T
- These Y_i ‘s (or Z_i’s) contribute at least a 1 to each 3

Related Problems

PARTITION

- PARTITION = {S : S finite subset of $\mathbb{Z}$ that can be split into two sets which sum to equal values}

- SUBSET-SUM $\leq_p$PARTITION

- So, PARTITION is NP-complete

KNAPSACK