Knapsack

* What is the knapsack problem?

* Say, there is a large no. of items (let there be k items of volume $v_i = 0, 1, 2 -- k-1$

Let there be a knapsack of volume V

The knapsack problem states that: Find a subset of k items ($I \subset \{1, 2 -- k\}$) such that

$$\boxed{\sum_{i \in I} v_i = V}$$, if such a subset exists

* Knapsack problem in case of integers are:

"Given a set of $\{v_i\}$ of k positive integers and an integer V, find a k-bit integer $n = (\epsilon_{k-1} \epsilon_{k-2} -- \epsilon_0)_2$ where $\epsilon_i \in \{0, 1\}$ are the binary digits of n, such that:

$$\boxed{\sum_{i=0}^{k-1} \epsilon_i v_i = V}$$ if such an n exists

* **superincreasing sequence** = let $\{x_1 - - x_n\}$ be a sequence. This sequence is superincreasing if each x_i is greater than the sum of all the previous x_i's.

Ex: $\{2, 3, 7, 15, 31\}$

$3 > 2$
$7 > 2 + 3$
$15 > 2 + 3 + 7$
$31 > 2 + 3 + 7 + 15$

* The general knapsack problem is in a very difficult class of problems called the "NP-complete" problems. It is equivalent in difficulty to the Travelling salesman Problem.

NP-complete problems = For these problems, the solution can be guessed and verified in polynomial time and here non-deterministic simply means no specific rule followed to make the guess

Algorithm to solve the knapsack Problem for a given superincreasing k-tuple & integer V:

(1) set w equal to V and set $j = k$

(2) starting with ϵ_{j-1} and decreasing the index of ϵ, choose all of the ϵ_i equal to 0 until you get to the first i - say i_0 such that $v_{i_0} \leq w$ and set $\epsilon_{i_0} = 1$

(3) replace w by $w - v_{i_0}$, set $j = i_0$; and if $w > 0$ go back to ~~set~~ step 2

(4) if $w = 0$, we're done: if $w > 0$ and all of the remaining $v_i > w$, then there is no solution $n = (\epsilon_{k-1} \cdots \epsilon_0)_2$ to the problem.

NOTE: The solution to the knapsack problem is unique

Ex: let $v_i = \{2, 3, 7, 15, 31\}$ and let $V = 24$.

Soln: Applying the general knapsack algorithm

$$\begin{array}{cccccc} & \epsilon_0 & \epsilon_1 & \epsilon_2 & \epsilon_3 & \epsilon_4 \\ v_i = \{ & 2, & 3, & 7, & 15, & 31 \} \end{array}$$

↑ start from here $(j = 4)$

At $j = 4$:

since $31 > 24 \Rightarrow \epsilon_4 = 0$

At $j = 3$:

since $15 < 24 \Rightarrow \epsilon_3 = 1$

replace $V = 24$ by $V = 24 - 15$
$= 9$

$V = 9$

at $j = 2$:

Now, $7 < \cancel{24}\ 9 \Rightarrow \epsilon_2 = 1$

replace V by $V = 9 - 7 = 2$

$V = 2$

at $j = 1$:

Now $3 > 2 \Rightarrow \epsilon_1 = 0$

at $j = 0$:

Now $2 <= 2 \Rightarrow \epsilon_0 = 1$

replace V by $V = 2 - 2 = 0$

$V = 0$

Since V becomes 0, the solution is:

$$(E_4 E_3 E_2 E_1 E_0)_2$$

$$= (01101)_2$$

$$= 1 + 4 + 8 = (13)_{10}$$

Merkle-Hellman system

This is one of the earliest public key cryptosystems.
It makes use of the knapsack problem on superincreasing tuple.
The Algorithm for encryption and decryption parts are:

Algorithm for key generation:

1) Choose a block size n. Integers upto n bits can be encrypted with this key.

2) Choose a random -superincreasing sequence of n-positive integers
$$W = (w_1, w_2 - - - - w_n)$$

3) choose a random integer q such that:

$$\boxed{q > \sum_{i=1}^{n} w_i}$$

4) choose a random integer r such that $\boxed{\gcd(r, q) = 1}$ {r and q are co-prime}

5) calculate the sequence:

$$B = (b_1, b_2 - - b_n)$$

where $\boxed{b_i = r w_i \bmod q}$

The public key = B and private key = (W, q, r)

Algo. for Encryption:

let m be n-bit message consisting of bits $m_1, m_2 - - m_n$. select each b_i for which m_i is non-zero and add them together. Equivalently calculate:

$$\boxed{c = \sum_{i=1}^{n} m_i b_i}$$

$\boxed{c = \text{ciphertext}}$

Algo. for Decryption:

To decrypt a ciphertext c, find the subset of B that sums to c. We do this by transforming the problem into one of finding a subset of W. This problem can be solved in polynomial time since W is superincreasing.

1) Calculate the modular inverse of r mod q ~~vvvvvv~~

$$r' = r^{-1} \bmod q$$

2) $c' = c r' \bmod q$

3) Solve the subset-sum problem for c' using the superincreasing sequence W by a simple greedy algorithm:

Let $X = (x_1, x_2 \dots x_k)$ be the resulting list of indexes of W that sum to c'.

$$\boxed{c' = \sum_{i=1}^{k} w_{x_i}}$$

4) Construct the msg. m with a 1 in each x_i bit position and 0 in all other bit positions.

$$\boxed{m = \sum_{i=1}^{k} 2^{n-x_i}}$$

The process can be understood from the example below:

Ex: Let the 8 bit message be $m = 97$.

Soln: Key generation: Create a key to encrypt 8 bit numbers.

$W = (2, 7, 11, 21, 42, 89, 180, 354)$

↳ random super-increasing sequence

The sum $\sum_{i=1}^{n} w_i = 706$, select a larger q

$$\boxed{q = 881}$$

Choose r co-prime to q.

$$\boxed{r = 588}$$

$32 + 64 + 1 = 65 + 32 = 97$

constructing $B = (b_1, b_2 - - b_n)$, $b_i = r w_i \bmod q$

$b_1 = (588 \times 2) \bmod 881 = 295$

$b_2 = (588 \times 7) \bmod 881 = 592$

$b_3 = (588 \times 11) \bmod 881 = 301$

$b_4 = (588 \times 21) \bmod 881 = 14$

$b_5 = (588 \times 42) \bmod 881 = 28$

$b_6 = (588 \times 89) \bmod 881 = 353$

$b_7 = (588 \times 180) \bmod 881 = 120$

$b_8 = (588 \times 354) \bmod 881 = 236$

$B = (295, 592, 301, 14, 28, 353, 120, 236)$

At encryption side:

$m = 97 = (01100001)_2$

$$C = \sum_{i=1}^{n} m_i b_i$$

$$= 0 \times 295 + 1 \times 592 + 1 \times 301 + 0 \times 14 + 0 \times 28 + 0 \times 353 + 0 \times 120 + 1 \times 236$$

$$\boxed{C = 1129}$$

At decryption side:

$r' = r^{-1} \bmod q = 588^{-1} \bmod 881 = 442$

$((588 \times 442) \bmod 881 = 1)$

$c' = cr' \bmod q = (1129 \times 442) \bmod 881 = 372$

using greedy algo., decompose 372 into sum of w_i values:

$w = (2, 7, 11, 21, 42, 89, 180, 354)$

$c' = 372$

$w_8 = 354 \le 372$

$c' = 372 - 354 = 18$

$w_3 = 11 \le 18$

$c' = 18 - 11 = 7$

$w_2 = 7 \le 7$

$c' = 7 - 7 = 0$

thus, $372 = 354 + 11 + 7 = w_8 + w_3 + w_2$

$\boxed{x = (8, 3, 2)}$

$$m = \sum_{i=1}^{3} 2^{n-x_i} = 2^{8-8} + 2^{8-3} + 2^{8-2}$$
$$= 2^0 + 2^5 + 2^6 = 1 + 32 + 64 = 97$$

$\boxed{m = 97}$