- 教材讨论
  - JH第2章第3节第1小节、第2小节的三个定义

问题1：字母表、词、语言

- alphabet、symbol、word、language
  - 它们是如何被形式化定义的？
  - 你能举出一些实际生活中的例子吗？

- 你能设计一种语言来编码全班同学问求的期末考试成绩吗？
  - 你设计的字母表、词和语言分别是什么？
- 你能设计一种语言来编码图片吗？
  - 你设计的字母表、词和语言分别是什么？
- 编码视频呢？
  - 你设计的字母表、词和语言分别是什么？

- 什么是concatenation of word？
  你能利用这个概念来定义这些新概念吗？
  - prefix/suffix/subword
  - concatenation of language

问题1：字母表、词、语言 (续)

Definition 2.3.1.10. *Let $\Sigma = \{s_1, s_2, \dots, s_m\}$, $m \geq 1$, be an alphabet, and let $s_1 < s_2 < \cdots < s_m$ be a linear ordering on Σ. We define the* **canonical ordering** *on Σ^* as follows. For all $u, v \in \Sigma^*$,*

$$\begin{aligned} u < v \ \text{ if } & |u| < |v| \\ & \text{or } |u| = |v|, u = xs_iu', \text{ and } v = xs_jv' \\ & \quad \text{for some } x, u', v' \in \Sigma^*, \text{ and } i < j. \end{aligned}$$

- 我们为什么需要一种词的排序规则？
- 你能解释这条排序规则吗？
- 你能给出一种不同的排序规则吗？

问题2：判定和优化问题

Definition 2.3.2.1. *A* **decision problem** *is a triple* (L, U, Σ) *where* Σ *is an alphabet and* $L \subseteq U \subseteq \Sigma^*$. *An algorithm* A **solves (decides)** *the decision problem* (L, U, Σ) *if, for every* $x \in U$,

(i) $A(x) = 1$ *if* $x \in L$, *and*
(ii) $A(x) = 0$ *if* $x \in U - L$ $(x \notin L)$.

- decision problem中的三个符号分别表示什么意思？
  - 这里的word是什么？
- 判定算法应该给出怎样的结果？

An equivalent form of a description of a decision problem is the following form that specifies the input-output behavior.

Problem $(\boldsymbol{L}, \boldsymbol{U}, \boldsymbol{\Sigma})$

Input: An $x \in U$.
Output: "yes" if $x \in L$,
"no" otherwise.

For many decision problems (L, U, Σ) we assume $U = \Sigma^*$. In that case we shall use the short notation $(\boldsymbol{L}, \boldsymbol{\Sigma})$ instead of (L, Σ^*, Σ).

- 你理解这段话的含义了吗？

问题2：判定和优化问题 (续)

- 这些判定问题分别是什么含义？它们的L分别是什么？
  - Primality testing $\{w \in \{0,1\}^* \mid Number(w) \text{ is a prime}\}$
  - Equivalence problem for polynomials
  - Satisfiability problem $\{w \in \Sigma_{logic}^+ \mid w \text{ is a code of a satisfiable formula in CNF}\}$
  - Clique problem $\{x\#w \in \{0,1,\#\}^* \mid x \in \{0,1\}^* \text{ and } w \text{ represents a graph that contains a clique of size } Number(x)\}$
  - Vertex cover problem $\{u\#w \in \{0,1,\#\}^+ \mid u \in \{0,1\}^+ \text{ and } w \text{ represents a graph that contains a vertex cover of size } Number(u)\}$
  - Hamiltonian cycle problem $\{w \in \{0,1,\#\}^* \mid w \text{ represents a graph that contains a Hamiltonian cycle}\}$
  - Existence of a solution of linear integer programming
    $\{\langle A, b\rangle \in \{0,1,\#\}^* \mid Sol_{\mathbb{Z}}(A,b) \neq \emptyset\}$

问题2：判定和优化问题 (续)

Definition 2.3.2.2. *An* **optimization problem** *is a 7-tuple* $U = (\Sigma_I, \Sigma_O, L, L_I, \mathcal{M}, cost, goal)$, *where*

(i) Σ_I *is an alphabet, called the* **input alphabet** *of* U,
(ii) Σ_O *is an alphabet, called the* **output alphabet** *of* U,
(iii) $L \subseteq \Sigma_I^*$ *is the* **language of feasible problem instances**,
(iv) $L_I \subseteq L$ *is the* **language of the (actual) problem instances of** U,
(v) $\mathcal{M}$ *is a function from* L *to* $Pot(\Sigma_O^*)$,[30] *and, for every* $x \in L$, $\mathcal{M}(x)$ *is called the* **set of feasible solutions** *for* x,
(vi) *cost is the* **cost function** *that, for every pair* (u, x), *where* $u \in \mathcal{M}(x)$ *for some* $x \in L$, *assigns a positive real number* $cost(u, x)$,
(vii) $goal \in \{minimum, maximum\}$.

- optimization problem中的七个符号分别表示什么意思？

An algorithm A *is* **consistent** *for* U *if, for every* $x \in L_I$, *the output* $A(x) \in \mathcal{M}(x)$. *We say that an algorithm* B **solves** *the optimization problem* U *if*

(i) B *is consistent for* U, *and*
(ii) *for every* $x \in L_I$, $B(x)$ *is an optimal solution for* x *and* U.

- 优化算法应该给出怎样的结果？

问题2：判定和优化问题 (续)

- 你能简述minimum vertex cover problem的含义吗？

Input: A graph $G = (V, E)$.
Constraints: $\mathcal{M}(G) = \{S \subseteq V \mid \text{every edge of } E \text{ is incident to at least one vertex of } S\}$.
Cost: For every $S \in \mathcal{M}(G)$, $cost(S, G) = |S|$.
Goal: *minimum*.

- 它和判定问题中的vertex cover problem之间有什么联系？

$$\{u\#w \in \{0,1,\#\}^+ \mid u \in \{0,1\}^+ \text{ and } w \text{ represents a graph that contains a vertex cover of size } Number(u)\}$$

问题2：判定和优化问题 (续)

- 你能简述maximum clique problem的含义吗？

Input: A graph $G = (V, E)$

Constraints: $\mathcal{M}(G) = \{S \subseteq V \mid \{\{u, v\} \mid u, v \in S, u \neq v\} \subseteq E\}$.

$\{\mathcal{M}(G)$ contains all complete subgraphs (cliques) of $G\}$

Costs: For every $S \in \mathcal{M}(G)$, $cost(S, G) = |S|$.

Goal: *maximum*.

- 它和判定问题中的clique problem之间有什么联系？

$$\{x\#w \in \{0,1,\#\}^* \mid x \in \{0,1\}^* \text{ and } w \text{ represents a graph that contains a clique of size } Number(x)\}$$

问题2：判定和优化问题 (续)

- 你能简述maximum cut problem的含义吗？

Input: A graph $G = (V, E)$.
Constraints:
$$\mathcal{M}(G) = \{(V_1, V_2) \mid V_1 \cup V_2 = V, V_1 \neq \emptyset \neq V_2, \text{and} V_1 \cap V_2 = \emptyset\}.$$
Costs: For every cut $(V_1, V_2) \in \mathcal{M}(G)$,
$$cost((V_1, V_2), G) = |E \cap \{\{u, v\} \mid u \in V_1, v \in V_2\}|.$$
Goal: *maximum*.

- 你能给出一个与之相关的判定问题吗？

问题2：判定和优化问题 (续)

- 你能简述traveling salesperson problem的含义吗？

Input: A weighted complete graph (G, c), where $G = (V, E)$ and $c : E \to \mathbb{N}$. Let $V = \{v_1, \ldots, v_n\}$ for some $n \in \mathbb{N} - \{0\}$.

Constraints: For every input instance (G, c), $\mathcal{M}(G, c) = \{v_{i_1}, v_{i_2}, \ldots, v_{i_n}, v_{i_1} \mid (i_1, i_2, \ldots, i_n)$ is a permutation of $(1, 2, \ldots, n)\}$, i.e., the set of all Hamiltonian cycles of G.

Costs: For every Hamiltonian cycle $H = v_{i_1} v_{i_2} \ldots v_{i_n} v_{i_1} \in \mathcal{M}(G, c)$, $cost((v_{i_1}, v_{i_2}, \ldots v_{i_n}, v_{i_1}), (G, c)) = \sum_{j=1}^{n} c(\{v_{i_j}, v_{i_{(j \bmod n)+1}}\})$, i.e., the cost of every Hamiltonian cycle H is the sum of the weights of all edges of H.

Goal: *minimum*.

问题2：判定和优化问题 (续)

- 你能简述knapsack problem的含义吗？

Input: A positive integer b, and $2n$ positive integers $w_1, w_2, \ldots, w_n$, c_1, $c_2, \ldots, c_n$ for some $n \in \mathbb{N} - \{0\}$.

Constraints:
$\mathcal{M}(b, w_1, \ldots, w_n, c_1, \ldots, c_n) = \{T \subseteq \{1, \ldots, n\} \mid \sum_{i \in T} w_i \leq b\}$.

Costs: For each $T \in \mathcal{M}(b, w_1, \ldots, w_n, c_1, \ldots, c_n)$,

$$cost(T, b, w_1, \ldots, w_n, c_1, \ldots, c_n) = \sum_{i \in T} c_i.$$

Goal: *maximum*.

问题2：判定和优化问题 (续)

- 你能简述bin-packing problem的含义吗？

Input: n rational numbers $w_1, w_2, \ldots, w_n \in [0,1]$ for some positive integer n.

Constraints: $\mathcal{M}(w_1, w_2, \ldots, w_n) = \{S \subseteq \{0,1\}^n \mid$ for every $s \in S$, $s^{\mathsf{T}} \cdot (w_1, w_2, \ldots, w_n) \leq 1$, and $\sum_{s \in S} s = (1, 1, \ldots, 1)\}$.
{If $S = \{s_1, s_2, \ldots, s_m\}$, then $s_i = (s_{i1}, s_{i2}, \ldots, s_{in})$ determines the set of objects packed in the ith bin. The jth object is packed into the ith bin if and only if $s_{ij} = 1$. The constraint

$$s_i^{\mathsf{T}} \cdot (w_1, \ldots, w_n) \leq 1$$

assures that the ith bin is not overfilled. The constraint

$$\sum_{s \in S} s = (1, 1, \ldots, 1)$$

assures that every object is packed in exactly one bin.}

Cost: For every $S \in \mathcal{M}(w_1, w_2, \ldots, w_n)$,

$$cost(S, (w_1, \ldots, w_n)) = |S|.$$

Goal: *minimum*.

问题2：判定和优化问题 (续)

- 你能简述makespan scheduling problem的含义吗？

Input:	Positive integers $p_1, p_2, \ldots, p_n$ and an integer $m \geq 2$ for some $n \in \mathbb{N} - \{0\}$. $\{p_i$ is the processing time of the ith job on any of the m available machines$\}$.
Constraints:	For every input instance $(p_1, \ldots, p_n, m)$ of MS, $\mathcal{M}(p_1, \ldots, p_n, m) = \{S_1, S_2, \ldots, S_m \mid S_i \subseteq \{1, 2, \ldots, n\}$ for $i = 1, \ldots, m$, $\bigcup_{k=1}^{m} S_k = \{1, 2, \ldots, n\}$, and $S_i \cap S_j = \emptyset$ for $i \neq j\}$. $\{\mathcal{M}(p_1, \ldots, p_n, m)$ contains all partitions of $\{1, 2, \ldots, n\}$ into m subsets. The meaning of $(S_1, S_2, \ldots, S_m)$ is that, for $i = 1, \ldots, m$, the jobs with indices from S_i have to be processed on the ith machine$\}$.
Costs:	For each $(S_1, S_2, \ldots, S_m) \in \mathcal{M}(p_1, \ldots, p_n, m)$, $cost((S_1, \ldots, S_m), (p_1, \ldots, p_n, m)) = \max\left\{\sum_{l \in S_i} p_l \mid i = 1, \ldots, m\right\}$.
Goal:	*minimum*.

问题2：判定和优化问题 (续)

- 你能简述set cover problem的含义吗？

Input: $(X, \mathcal{F})$, where X is a finite set and $\mathcal{F} \subseteq Pot(X)$ such that $X = \bigcup_{S \in \mathcal{F}} S$.
Constraints: For every input $(X, \mathcal{F})$,
$\mathcal{M}(X, \mathcal{F}) = \{C \subseteq \mathcal{F} \mid X = \bigcup_{S \in C} S\}$.
Costs: For every $C \in \mathcal{M}(X, \mathcal{F})$, $cost(C, (X, \mathcal{F})) = |C|$.
Goal: $minimum$.

问题2：判定和优化问题 (续)

- 你能简述maximum satisfiability problem的含义吗？

Input: A formula $\Phi = F_1 \wedge F_2 \wedge \cdots \wedge F_m$ over $X = \{x_1, x_2, \ldots\}$ in CNF (an equivalent description of this instance of MAX-SAT is to consider the set of clauses $F_1, F_2, \ldots, F_m$).

Constraints: For every formula Φ over the set $\{x_1, \ldots, x_n\} \subseteq X$, $n \in \mathbb{N} - \{0\}$, $\mathcal{M}(\Phi) = \{0, 1\}^n$.
{Every assignment of values to $\{x_1, \ldots, x_n\}$ is a feasible solution, i.e., $\mathcal{M}(\Phi)$ can also be written as $\{\alpha \mid \alpha : X \to \{0, 1\}\}$.

Costs: For every Φ in CNF, and every $\alpha \in \mathcal{M}(\Phi)$, $cost(\alpha, \Phi)$ is the number of clauses satisfied by α.

Goal: *maximum*.

问题2：判定和优化问题 (续)

- 你能简述integer linear programming的含义吗？

Input: An $m \times n$ matrix $A = [a_{ij}]_{i=1,\ldots,m,j=1,\ldots,n}$, and two vectors $b = (b_1, \ldots, b_m)^\mathsf{T}$, $c = (c_1, \ldots, c_n)^\mathsf{T}$ for some $n, m \in \mathbb{N} - \{0\}$, a_{ij}, b_i, c_j are integers for $i = 1, \ldots, m$, $j = 1, \ldots, n$.

Constraints: $\mathcal{M}(A, b, c) = \{X = (x_1, \ldots, x_n) \in \mathbb{Z}^n \mid AX = b$ and $x_i \geq 0$ for $i = 1, \ldots, n\}$.

Costs: For every $X = (x_1, \ldots, x_n) \in \mathcal{M}(A, b, c)$, $cost(X, (A, b, c)) = \sum_{i=1}^{n} c_i x_i$.

Goal: *minimum*.

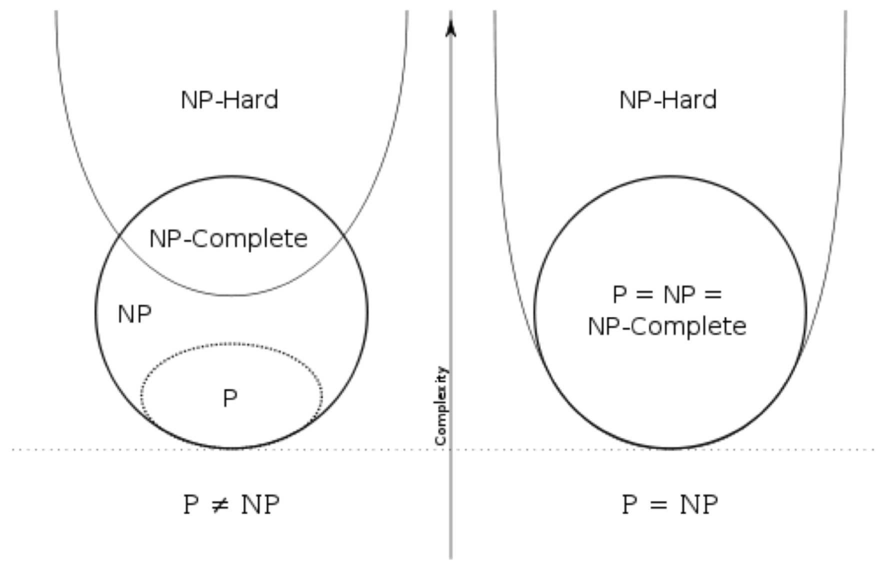
问题2：判定和优化问题 (续)

- 你能简述maximum linear equation problem mod *k*的含义吗？

Input:	A set S of m linear equations over n unknowns, $n, m \in \mathbb{N} - \{0\}$, with coefficients from $\mathbb{Z}_k$. (An alternative description of an input is an $m \times n$ matrix over $\mathbb{Z}_k$ and a vector $b \in \mathbb{Z}_k^m$).
Constraints:	$\mathcal{M}(S) = \mathbb{Z}_k^m$ $\{$a feasible solution is any assignment of values from $\{0, 1, \ldots, k-1\}$ to the n unknowns (variables)$\}$.
Costs:	For every $X \in \mathcal{M}(S)$, $cost(X, S)$ is the number of linear equations of S satisfied by X.
Goal:	*maximum*.

问题3：P和NP

- 你能解释清楚这些概念及其之间的关系吗？
  - P
  - NP
  - NP-hard
  - NP-complete



- 注意，经常被忽视的一点是，严格来说
  - P、NP、NP-complete都是描述判定问题的
  - NP-hard可以描述判定问题、优化问题等各类问题

问题3：P和NP (续)

- 优化问题也有自己的“NP”和“P”，你理解了吗？

Definition 2.3.3.21. NPO *is the class of optimization problems, where* $U = (\Sigma_I, \Sigma_O, L, L_I, \mathcal{M}, cost, goal) \in \mathrm{NPO}$ *if the following conditions hold:*

(i) $L_I \in \mathrm{P}$,
(ii) there exists a polynomial p_U *such that*
  a) for every $x \in L_I$*, and every* $y \in \mathcal{M}(x)$, $|y| \le p_U(|x|)$*, and*
  b) there exists a polynomial-time algorithm that, for every $y \in \Sigma_O^*$ *and every* $x \in L_I$ *such that* $|y| \le p_U(|x|)$*, decides whether* $y \in \mathcal{M}(x)$*, and*
(iii) the function cost is computable in polynomial time.

Informally, we see that an optimization problem U is in NPO if

(i) one can efficiently verify whether a string is an instance of U,
(ii) the size of the solutions is polynomial in the size of the problem instances and one can verify in polynomial time whether a string y is a solution to any given input instance x, and
(iii) the cost of any solution can be efficiently determined.

Definition 2.3.3.23. PO *is the class of optimization problems* $U = (\Sigma_I, \Sigma_O, L, L_I, \mathcal{M}, cost, goal)$ *such that*

(i) $U \in \mathrm{NPO}$*, and*
(ii) there is a polynomial-time algorithm that, for every $x \in L_I$*, computes an optimal solution for* x.