Недетерміновані алгоритми.

Питання

- Недетерміновані машини Тьюринга.
- Класи NTIME, NP і co-NP.
- Альтернативне визначення класу NP.
- Класи з ємнісною складності.

- ***k-стрічковою недетермінованою машиною Тьюринга*** називається п'ятірка $M = (A, Q, S, \Pi, q_0)$.

- Значення всіх компонент п'ятірки - такі ж, як і в разі k-стрічкової детермінованої МТ, за винятком того, що програма П являє собою не відображення, а відношення, задане на множині $(Q\times A^k)\times(Q\times(A\times S)^k)$.

- Принцип роботи НМТ в цілому такий же, як і у ДМТ. Але для деяких конфігурацій машини (наборів $(Q, a_1, ..., a_k)$) може існувати кілька конфігурацій $(q', a'_1, ..., a'_k, s'_1, ..., s'_k)$, пов'язаних ставленням П з поточної конфігурацій, тобто таких, що $(q, a_1, ..., a_k, q', a'_1, ..., a'_k, s'_1, ..., s'_k) \in \Pi$. В цьому випадку машина в якості наступної конфігурації вибирає *будь-який* такий набір.

- НМТ *M* *допускає* вхідне слово *x*, якщо *хоча б одна* така послідовність конфігурацій призводить до стану q^Y, що допускається (така послідовність називається *той, що допускає)*.

- Якщо *жодна* послідовність змін не призводить до стану, що допускається, машина *M* *відхиляє* слово *x*.

- Таким чином, на відміну від детермінованих машин, ситуації допуску або відхилення вхідного слова не симетричні.

- НМТ М має ***часову складність* T(n),** якщо для будь-якого вхідного слова, що допускається, довжини n знайдеться послідовність, яка складається не більше ніж з T(n) кроків та приводить в стан, що допускається.

- НМТ М має **місткy складність S(n),** якщо для будь-якого вхідного слова, що допускається, довжини n знайдеться послідовність, яка веде в стан, що допускається, в якій число переглянутих комірок на кожній стрічці не перевищує S(n).

- **ТЕОРЕМА 1.** Для будь-якої k-стрічкової недетермінованої машини М, що має тимчасову складність T(n), існує одно стрічкова НМТ М', що моделює М з часовою складністю $T'(n)=O(T^2(n))$.

- Масова проблема називається ***алгоритмічно розв'язуваною***, якщо існує алгоритм (машина Тьюрінга), який дозволяє розв'язати кожну задачу цієї проблеми і ***алгоритмічно нерозв'язуваною***, якщо такого алгоритму не існує.

- **Розмір задачі** – це об'єм вхідних даних, необхідних для завдання всіх параметрів задачі.

- Час, що витрачається алгоритмом, як функція розміру задачі називається ***часовою складністю*** цього алгоритму, позначається T(n). Поведінка цієї складності при граничному переході при збільшенні розміру задачі називається ***асимптотичною часовою складністю***. Аналогічно можна визначити місткісну складність S(n) і ***асимптотичну місткісну складність.***

Класифікація алгоритмів за їх часовою складністю

1. Алгоритм називається ***сталим***, якщо його часова складність не залежить від *n*.

2. Алгоритм називається ***поліноміальним***, якщо його часова складність дорівнює:

$$O(n^m): T(n) = O(n^m), \text{ де } m = const,$$

тобто робоча функція поліноміального алгоритму є многочленом степеня *m*.

3. Алгоритм називається ***експоненціальним***, якщо $T(n) = O(t^{g(n)})$, де $t > 1$

де $t>1$ – константа, *g(n)* – деяка поліноміальна функція.

4. Алгоритм називається ***суперполіноміальним***, якщо $T(n) = O(c^{g(n)})$,

де із зростанням *n* функція *g (n)* зростає швидше константи *c*.

Клас алгоритмів	Складність $T(n)$	Кількість операцій процесора при $n=10^6$	Час зі швидкістю $n=10^6$ операцій за секунду
Сталі	$O(1)$	1	1 мкс
Лінійні	$O(n)$	10^6	1 с
Квадратичні	$O(n^2)$	10^{12}	11,6 доби
Кубічні	$O(n^3)$	10^{18}	32 000 років
Експоненціальні	$O(2^n)$	10^{301030}	У 10^{301006} разів довше за час існування Всесвіту

- **Розв’язуваними** називаються задачі, в яких використовуються алгоритми з поліноміальною часовою складністю.

- **Складнорозв’язуваними** або просто **складними** називаються задачі, які неможливо розв’язати за поліноміальний час.

- **Нерозв’язуваними** називаються задачі, для розв’язування яких не можна побудувати жодного алгоритму, навіть без урахування часової складності.

Класифікація задач за їх складністю розв’язування

- ***Клас*** *P* утворюють задачі, які можна розв’язати за поліноміальний час.
- ***Клас*** *NP* утворюють задачі, які можна розв’язати за поліноміальний час тільки на недетермінованій машині Тьюрінга.
- Класи *P* і *NP* знаходяться між собою у відношенні нестрогого включення: $P \subseteq NP$. Проблема збіжності класів $P = NP$ або незбіжності $P \neq NP$ вважається в теорії складності центральною і досі не вирішена.
- ***Клас NP-повних задач*** складають такі задачі з класу *NP*, що з побудови поліноміального алгоритму для будь-якої такої задачі випливає можливість побудови такого ж алгоритму для всіх інших задач класу *NP*.
- ***Клас PSPACE*** утворюють задачі, які можна розв’язати у поліноміальному просторі, але необов’язково за поліноміальний час.
- ***Клас PSPACE -повних задач*** складають задачі, які мають властивість: якщо кожна з них є *NP*-задачею, то $PSPACE=NP$, а якщо *P*-задачею, то $PSPACE=P$.
- ***Клас EXPTIME*** утворюють задачі, які можна розв’язати за експоненціальний час.

Класи задач за складністю щодо недетермінірованних алгоритмів:

- **NDTIME(f(n))** - це клас задач, для кожної з яких існує недетерміровані одностірчкові МТ, що розв’язує цю задачу з часовою складністю O(f (n)).

- **NP=NPTIME=**$\cup_{k>0}$**NDTIME(**n^k**)** - клас задач, що розв’язуються недетермінованими алгоритмами за поліноміальний час.

- **NEXPTIME=**$\cup_{k>0}$**NDTIME(2^(**n^k**))** - клас задач, що розв’язується недетермінованими алгоритмами за експоненціальний час.

- **ТЕОРЕМА 2.** Для будь-якої НМТ M^N, яка розпізнає мову L за час T(n) знайдеться ДМТ M^D і константа c, такі, що M^D також розпізнає мову L, але за час $O(c^{T(n)})$.

Мова L належить класу NP тоді і тільки тоді, коли існує детермінована машина Тьюринга M і поліном p, такі що:

- для будь-якого слова $x \in L$ існує слово-сертифікат y, $|y| \leq p(|x|)$, при якому машина M допускає пару (x, y) за поліноміальний час;
- ні для якого $x \notin L$ подібного сертифіката не існує.

Класи з місткою складністю.

- ***DSPACE(f(n))*** - це клас мов, для кожного з яких існує детермінована МТ, що розпізнає цю мову з місткою складністю O(f(n)).

- **PSPACE=$\cup_{k>0}$DSPACE(n^k)** - клас мов, які розпізнаються з поліноміальною ємністю.

- **NSPACE(f(n))** - це клас мов, для кожного з яких існує недетермінірована МТ, що розпізнає цю мову з місткою складністю O(f(n)).

- **NPSPACE=$\cup_{k>0}$NSPACE(n^k)** - клас мов, які розпізнаються недетермінованими машинами з поліноміальною місткістю.

ТЕОРЕМА 4. PSPACE = NPSPACE

З теореми (для будь-якої машини Тьюрінга виконується нерівність S(n)≤T(n), тобто місткa складність не перевищує часову) слідує, що для будь-якої функції f справедливі вкладення DTIME(f(n))∈DSPACE(f(n)) і NTIME(f(n))∈NSPACE(f(n)).

Отже, NP∈ NPSPACE. Тому співвідношення розглянутих класів можна зобразити у вигляді такої схеми.