planetmath.org

Math for the people, by the people.

complexity class

Canonical name	ComplexityClass
Date of creation	2013-03-22 13:01:59
Last modified on	2013-03-22 13:01:59
Owner	Henry (455)
Last modified by	Henry (455)
Numerical id	4
Author	Henry (455)
Entry type	Definition
Classification	msc 68Q15

If $f(n)$ is any function and T is a Turing machine (of any kind) which halts on all inputs, we say that T is time bounded by $f(n)$ if for any input x with length $|x|$, T halts after at most $f(|x|)$ steps.

For a decision problem L and a class K of Turing machines, we say that $L \in KTIME(f(n))$ if there is a Turing machine $T \in K$ bounded by $f(n)$ which decides L. If R is a search problem then $R \in KTIME(f(n))$ if $L(R) \in KTIME(f(n))$. The most common classes are all restricted to one read-only input tape and one output/work tape (and in some cases a one-way, read-only guess tape) and are defined as follows:

- D is the class of deterministic Turing machines. (In this case $KTIME$ is written $DTIME$.)
- N is the class of non-deterministic Turing machines (and so $KTIME$ is written $NTIME$).
- R is the class of positive one-sided error Turing machines and coR the class of negative one-sided error machines
- BP is the class of two-sided error machines
- P is the class of minimal error machines
- ZP is the class of zero error machines

Although $KTIME(f(n))$ is a time complexity class for any $f(n)$, in actual use time complexity classes are usually the union of $KTIME(f(n))$ for many f. If Φ is a class of functions then $KTIME(\Phi) = \bigcup_{f \in \Phi} KTIME(f(n))$. Most commonly this is used when $\Phi = \mathcal{O}(f(n))$.

The most important time complexity classes are the polynomial classes:

$$K\mathcal{P} = \bigcup_{i \in \mathbb{N}} KTIME(n^i)$$

When $K = D$ this is called just $\mathcal{P}$, the class of problems decidable in polynomial time. One of the major outstanding problems in mathematics is the question of whether $\mathcal{P} = \mathcal{NP}$.

We say a problem $\pi \in KSPACE(f(n))$ if there is a Turing machine $T \in K$ which solves π, always halts, and never uses more than $f(n)$ cells of its output/work tape. As above, if Φ is a class of function then $KSPACE(\Phi) = \bigcup_{f \in \Phi} KSPACE(f(n))$

The most common space complexity classes are $K\mathcal{L} = KSPACE(\mathcal{O}(\log n))$. When $K = D$ this is just called $\mathcal{L}$.

If $\mathcal{C}$ is any complexity class then $\pi \in co\mathcal{C}$ if π is a decision problem and $\overline{\pi} \in \mathcal{C}$ or π is a search problem and $L(\pi) \in co\mathcal{C}$. Of course, this coincides with the definition of coR above. Clearly $co(co\mathcal{C}) = \mathcal{C}$.

Since a machine with a time complexity $f(n)$ cannot possibly use more than $f(n)$ cells, $KTIME(f(n)) \subseteq KSPACE(f(n))$. If $K \subseteq K'$ then $KTIME(f(n)) \subseteq K'TIME(f(n))$ and similarly for space.

The following are all trivial, following from the fact that some classes of machines accept and reject under stricter circumstances than others:

$$D \subseteq ZP = R \cap coR$$

$$R \cup coR \subseteq BP \cap N$$

$$BP \subseteq P$$