

Algoritmická řešitelnost problémů

Z FAV wiki

Problém definujeme jako binární relaci mezi množinou instancí problému I (tj. množinou všech možností vstupu) a množinou řešení S. Dvě instance mohou mít stejné řešení, stejně tak jedna může mít více řešení. Algoritmická řešitelnost zkoumá, zda pro všechny formulovatelné problémy lze nalézt algoritmus řešení.

Ve 30. l. 20. st. objevil Alan Turing formální opis algoritmu - Turinguv stroj, ve spojení s Alonzem Churchem vytvořili Churchovu-Turingovu tezi a sice, že každý algoritmus (ne problém!) lze vykonat Turingovým strojem. Tezi lze vyvrátit nalezením algoritmu nevykonatelného TS. Moderní programovací jazyky pak byly navrženy tak, aby libovolný program šlo převést na TS a naopak. Problém, který nelze vyřešit pomocí TS, tedy ani pomocí programu, je pak algoritmicky nerešitelný.

Vytvoríme-li takový rozhodovací problém (řešení je ano/ne), který pro každý rozhodovací algoritmus a alespon jeden jeho vstup dává pro tento vstup opakovanou odpověď, než daný algoritmus se stejným vstupem, takový problém není řešitelný žádným z těchto algoritmu a je tedy nerozhodnutelný.

První takový problém - problém zastavení - našel sám Turing: program má o všech vytvořitelných programech rozhodnout, zda pro každý ze vstupu daný program zastaví (tehdy vrátí jeho výstup v podobě přirozeného čísla + 1) nebo nezastaví (pak vrátí 0). Takový program to ale nedokáže rozhodnout sám o sobě (pokud by byl vytvořitelný, patřil by mezi zkoumané programy také a musel by ve svém výstupu vrátit svůj výstup + 1, což nejde).

Citováno z „[http://www.512.cz/index.php?](http://www.512.cz/index.php?title=Algoritmick%C3%A1_%C5%99e%C5%A1itelnost_prob%C3%A9m%C5%AF)

[title=Algoritmick%C3%A1_%C5%99e%C5%A1itelnost_prob%C3%A9m%C5%AF](http://www.512.cz/index.php?title=Algoritmick%C3%A1_%C5%99e%C5%A1itelnost_prob%C3%A9m%C5%AF)“

Kategorie: Fav-kiv-bzinf

- Stránka byla naposledy editována 20. 2. 2014 v 06:37.
- Stránka byla zobrazena 820krát.