

Algoritmizace

Pojmy

- **Algoritmus** - přesný postup, kterým lze vyřešit určitou úlohu
- **Program** - zápis algoritmu v určitém programovacím jazyce (je spustitelný počítačem)
- **Programovací jazyk** - prostředek pro zápis algoritmů a tvorbu programů

Pravidla programovacích jazyků

- **Syntaxe** (forma) - Definuje kombinaci symbolů, které jsou považovány za správně strukturovaný kód v konkrétním jazyce (např. středník na konci řádku, řetězec v uvozovkách, použití různých typů závorek). Při porušení syntaktických pravidel nejde program zkompilovat.
- **Sémantika** (význam) - Zhodnocuje význam syntakticky platných řetězců a provádí jejich výpočty. Popisuje procesy, které provádí počítač při vykonávání programu (např. $a = 1 + 1$ - dojde k sečtení dvou čísel a výsledek bude uložen do proměnné s názvem „a“).

Vlastnosti algoritmus

- **Determinovanost** - v každé situaci musí být naprosto zřejmé, co a jak se má provést, jak má provádění algoritmu pokračovat, pro stejné vstupní data musí být stejný výstup
- **Obecnost** - algoritmus by neměl řešit jeden konkrétní problém (například 5×5), ale měl by nabízet obecné řešení daného problému (například $X \times Y$)
- **Finitivnost** - algoritmus by měl vždy mít omezený počet kroků, po kterých skončí
- **Resultativnost** - musí mít nějaký výstup, který je řešením daného úkolu
- **Korektnost** - výstup by měl být správně
- **Efektivita** - dělí se na paměťovou efektivitu (náročnost na paměť) a výpočetní efektivitu (náročnost na výpočet), tyto dvě vlastnosti jsou většinou k sobě ve vztahu nepřímé úměry

Znamé algoritmy

Eratostenovo síto

Algoritmus pro získání všech prvočísel od dvou po dané číslo. Vytvoříme si pole všech čísel obsažených v daném rozsahu. Postupujeme postupně přes všechna čísla rozsahu a odebíráme z něj čísla, která jsou násobky těchto čísel. Algoritmus končí pokud je z pole odebráno poslední číslo, nebo pokud je jako prvočíslo označeno číslo vyšší než odmocnina nejvyššího čísla (pak jsou všechny zbylé prvky prvočísla).

[Ukázka erastotenova síta na číslech od 2 do 120](#)

Euklidův algoritmus

Algoritmus pro výpočet největšího společného dělitele (dále jen NSD) dvou čísel.

Nejlépe se to asi ukáže na příkladu. Máme zadaná dvě čísla 140 a 15.

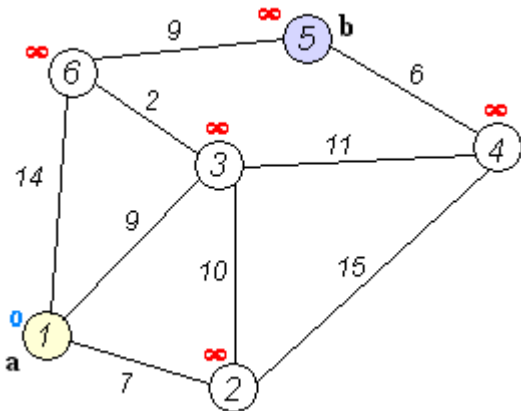
Postup:

- Nejprve zjistíme zbytek po dělení většího čísla číslem menším. (V našem případě $140 = 9 * 15 + 5$)
- Nyní zopakujeme první krok, ale s dělením menšího čísla zbytkem po dělení ($15 = 3 * 5 + 0$)
- Vyšel nám zbytek 0, takže NSD je rovno 5, pokud by nám nevyšel zbytek 0 aplikovali bychom znovu krok jedna

[Podrobnější vysvětlení](#)

Dijkstrův algoritmus

Algoritmus sloužící pro výběr nejlepší trasy z bodu A do bodu B. Funguje tak, že každá cesta mezi jednotlivými body dostane hodnotu, podle „náročnosti“ (délka trasy, povolená rychlost, ...). Algoritmus si postupně vypočítává délku cesty do všech sousedních bodů a z nich do dalších bodů. Pokud algoritmus najde novou cestu do již objeveného bodu, pomalejší cestu k tomuto bodu odstraní. Na konci zůstane pouze jedna nejrychlejší cesta do požadovaného bodu.



From:

<https://old.gml.cz/wiki/> - GMLWiki

Permanent link:

<https://old.gml.cz/wiki/doku.php/informatika:maturita:16a?rev=1552375448>

Last update: **12. 03. 2019, 08.24**

