



Aplikovaná informatika

Podklady předmětu
Aplikovaná informatika
pro akademický rok 2006/2007
Radim Farana





Obsah

- Tvorba algoritmů,
 - vlastnosti algoritmu.
- Popis algoritmů,
 - vývojové diagramy,
 - strukturogramy.
- Hodnocení složitosti algoritmů,
 - vypočitatelnost,
 - časová složitost,
 - NP a NP-úplné problémy.



Algoritmus

- **Algoritmus** je přesný předpis definující výpočtový proces vedoucí od měnitelných výchozích údajů až k žádaným (vždy správným) výsledkům. Tento předpis se skládá z jednotlivých výpočtových kroků, které jsou zapsány v určitém pořadí. Počet výpočtových kroků musí být konečný.



Vlastnosti algoritmu

- **determinovanost** - shrnuje **přesnost**, **srozumitelnost** a **jednoznačnost**. V každém okamžiku řešení musí být jasné, jakou operaci má algoritmus provádět.
- **hromadnost** (masovost) - algoritmus musí popisovat zpracování celé skupiny příbuzných hodnot.
- **rezultativnost** - algoritmus musí vždy dospět ke správnému výsledku, a to pomocí **konečného** počtu kroků.
- **opakovatelnost** - při stejných hodnotách vstupních dat musí algoritmus vždy dospět ke stejnému výsledku.



Algoritmus versus program

- **program** = posloupnost příkazů,
 - dokumentu je se výpisem programu,
 - je chráněn autorským zákonem.
- **algoritmus** = postup práce,
 - dokumentuje se zápisem algoritmu,
 - je možné ho patentovat.
- Program realizuje algoritmus (algoritmy), algoritmus je jeho nutnou součástí.



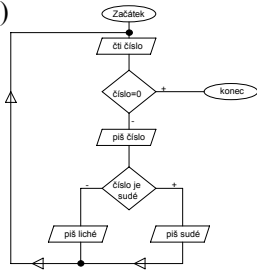
Popis algoritmu

- Slovní popis
 - pracovní postup,
 - strukturovaný text, zápis pomocí grafu,
 - pseudokód (programovací).
- Grafický zápis
 - vývojový diagram,
 - Kopenogram, NS-diagram,
 - strukturogram.



Vývojový diagram

- Popis algoritmů pro FORTRAN (FORMula TRANslator)
- IBM v r. 1954
- Formalizován různými normativy (ČSN 36 9030)



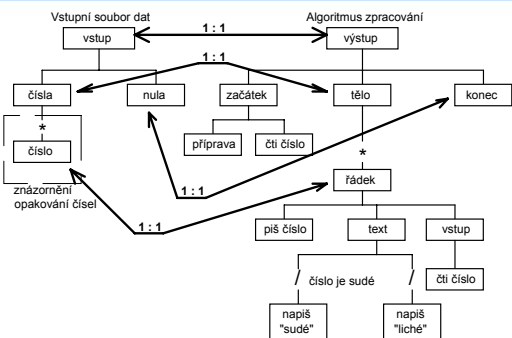


Strukturogram

- M. A. Jackson, 1975
- Základní struktury:
 - sekvence (posloupnost operací),
 - selekce (větvení).
 - opakování – zvláštní případ sekvence.
- Snadné postupné upřesňování algoritmu
- Jednoznačný vztah mezi daty a algoritmem



Strukturogram



Aplikovaná informatika 10

Základní struktury

Pseudojazyk	Vývojový diagram	Kopenogram	Strukturogram
Programový celek (rutina, podprogram, procedura a pod.) - definice			
název (parametry) . . end název		 název horní pruh vybarven žlutě	jeden celý strukturogram
programový celek - použití (volání)			
název (parametry)		 název blok vybarven červeně, při rekurzivním volání žlutě	

Aplikovaná informatika 11

Podmíněná činnost, rozhodování

Pseudojazyk	Vývojový diagram	Kopenogram	Strukturogram
Provedení konkrétní činnosti			
popis činnosti		 popis činnosti blok vybarven červeně	
Podmíněná činnost (provádí se pouze pokud je splněna určitá podmínka)			
if podmínka podmíněná činnost end if		 podmínka podmíněná činnost horní pruh vybarven modře	
Rozhodování (pokud platí určená podmínka, provede se činnost 1, jinak činnost 2)			
if podmínka činnost 1 else činnost 2 end if		 podmínka činnost 1 činnost 2 horní pruh vybarven modře	

Aplikovaná informatika 12

Větení (podle hodnoty výrazu se provádí určená činnost)

Pseudojazyk	Vývojový diagram	Kopenogram	Strukturogram
case výraz=hodnota 1 činnost 1 case výraz=hodnota 2 činnost 2 case else činnost při neznámé hodnotě end case		 výraz = hod. 1 výraz = hod. 2 činn. při nezn. hodnotě horní pruh vybarven modře	
Opakování s pevným počtem opakování			
for počtadlo=začátek to konec step krok činnost end for		 počet opakování činnost horní a dolní pruh vybarveny zeleně	



Opakování

Pseudojazyk	Vývojový diagram	Kopenogram	Strukturogram
Opakování s testem na začátku (dokud platí podmínka, činnost se opakuje - pokud na začátku opakování není podmínka splněna, činnost se vůbec neprovede)			
while podmínka činnost end while			
Opakování s testem na konci (opakování končí, pokud je splněna podmínka - i když podmínka platí již na začátku opakování, činnost se jednou provede)			
repeat činnost until podmínka (v některých jazycích while činnost end while podmínka)			neexistuje

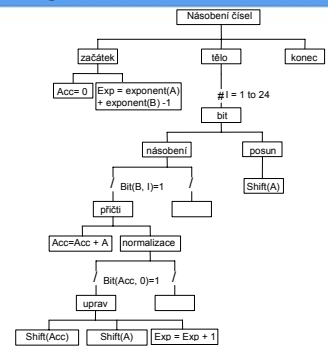


Zvláštní činnosti

Pseudojazyk	Vývojový diagram	Kopenogram	Strukturogram
Vstupní nebo výstupní operace			
Jako každá jiná činnost		Jako každá jiná činnost	Jako každá jiná činnost
Přípravná činnost			
Jako každá jiná činnost		Jako každá jiná činnost	Jako každá jiná činnost
Spojka (činnost končí v jedné části algoritmu a pokračuje v jiné části)			
Neexistuje		Neexistuje	Neexistuje



Algoritmus násobení dvou čísel Single



Proměnné:
 A, B – činitele
 (normalizované mantisy)
 Acc – akumulátor
 Exp – exponent
 I – celočíselné počítadlo
Funkce:
 Shift(x) – bitová rotace vpravo
 exponent(x) – exponent čísla

Aplikovaná informatika 16

Testování algoritmu

Proměnné:

A : 0,110011001100110011001100 · 2⁻³
 0,011001100110011001100110
 0,001100110011001100110011
 0,000110011001100110011001

B : 0,100011001100110011001100 · 2⁻³

Acc : 1,00110011001100110011001100
 1,0011001100110011001100110010

Exp: -6

I: 3

atd.

Chyba opisu, riziko ručního testování

Aplikovaná informatika 17

Hodnocení složitosti algoritmů

- Vypočitatelnost algoritmu
 - Turingův stroj (Alan M. Turing, 1936) – abstraktní model počítače,
 - s Alonzem Churchem vyslovili domněnku, že je ekvivalentní s počítačem (s nekonečnou pamětí),
 - Dokázal, že nelze sestavit stroj, který určí, zda se libovolný stroj zastaví.

Další krok závisí na vnitřním stavu a přečtené hodnotě

Aplikovaná informatika 18

Časové omezení

- Čas na vykonání algoritmu vyjádřený v počtu elementárních operací.
- Všechny operace trvají stejně dlouho.
- **Složitost** algoritmu je funkcí vstupu (n).
- Konstantní rozdíly se ignorují.
- Necht' $f(n)$, $g(n)$ jsou funkce z množiny přirozených čísel do množiny reálných čísel, pak $f(n) = O(g(n))$, pokud existuje konstanta c , že pro velké n platí: $f(n) \leq cg(n)$.



Třídy NP a NP-úplných problémů

- **P** – schůdné algoritmy běžící nejhůře v polynomiálním čase (PT) – násobení,
- **NP problém** (nondeterministic polynomial) – neschůdné algoritmy, v PT lze pouze ověřit správnost řešení – faktorizace ,
- **NP-úplný problém** – NP problémy vzájemně mapovatelné v PT – obchodní cestující, (existuje P transformace jednoho na druhý).
