

# Algoritmusok fogalma. Algoritmusok hatékonysága. Típusok fogalma. Alaptípusok, összetett típusok.

## Algoritmus fogalma:

Az algoritmus több azonos jellegű, egymástól csak a kiinduló adatokban különböző (pl. matematikai) feladat megoldására szolgáló eljárás, amelynek során előre meghatározott, véges számú lépést adott sorrendben végrehajtva jutunk el a feladat megoldásához.

## Algoritmus

- Recept
- Számítási eljárás
- Filozófia, ami leír
  - Megoldható feladatokat
  - Megfogalmazható problémákat
  - Kifejezhető gondolatokat
- jó program tervezése, egyértelmű lépések véges sorozata = megoldás
- független számítógéptől, programnyelvtől
- lehető legrövidebb, legtermészetesebb algoritmusra való törekvés fontos
- az algoritmus egy programnál az elsődleges

## Algoritmus specifikációja

Algoritmus: egy függvény, amely

- Bemenete: értelmezési tartomány (adatszerkezet)
- Kimenete: az értékkészlet (adatszerkezet)
- Specifikáció: a függvény definíciója, amely megadja, hogy milyen bemenethez milyen kimenetet kell előállítania.

## Algoritmus kifejezése/megfogalmazása

- Természetes nyelven
- Pszeudokóddal
- Blokkdiagrammal
- Tervezési nyelvvel (UML)
- Programnyelvvel

Adott specifikációhoz többféle megfogalmazás is létezhet.

## Algoritmusok elemzése - Hatékonyságvizsgálat

- Cél: az algoritmus futásához szükséges erőforrások/költség előrejelzése
  - Számítási/futási idő (elemi lépések száma)
  - Tárigény
  - Sávszélesség
  - Minimális processzor sebesség stb.
- Feladatok:
  - Legrosszabb eset (worst case)
  - Átlagos eset (várható érték)
  - Növekedési rend
- ciklusok végrehajtási száma (egymásba ágyazott ciklusok nagyon lassítják),
- egy ciklus végrehajtási ideje,
- feltételvizsgálat száma,
- kivételesek kiküszöbölése.

## **Tipikus hibák**

Futási idő alul becslése

Tárterület alul becslése

Fogalom: kombinatorikai robbanás (sakktábla): A lehetséges helyzetek száma a játék szintjeinek (lépéseinek) számával hatványozottan nő (exponential explosion).

## **Növekedési rend**

- Költség becslése (leggyakrabban) a bementi adatok (leggyakrabban méretének) függvényében
- Elsősorban nagyon nagy változóértékekre
- Konstans szorzótényezőtől függetlenül
- Legmagasabb kitevőjű tagra
- Jelölés (több is van):  $\Theta(f(\text{bemenet}))$
- Pl.  $\Theta(n^2)$ ,  $\Theta(\log(n))$ ,  $\Theta(n)$

## **Típusok**

- Halmaz (a világ objektumai közül néhány)
- Neve van
- Típushoz tartozási döntés (Real Time Type Information, RTTI)
- Részei lehetnek (Láthatóság!)
- Műveletei lehetnek
- Egyszerű típusok  $\leftarrow \rightarrow$  összetett típusok
- Alaptípusok
- Véges (korlátos méretű) típusok  $\leftarrow \rightarrow$  nem véges típusok
- Rekurzív típusok
- 
- Alaptípusok (skaláris típusok, nincsenek részei)
- Felsorolástípusok (SILK/SILAN, UML, Pascal, VB)
- enumeration Boolean {true, false};
- enumeration Tristate {true, false, error};

## **Boolean**

and(x : Boolean) : Boolean

or(x : Boolean) : Boolean

xor(x : Boolean) : Boolean

not() : Boolean

implies(impl : Boolean) : Boolean

ifthenelse(then : Boolean, else : Boolean) : Boolean

## **Integer**

'+'(x : Integer) : Integer

'-'(x : Integer) : Integer

'\*'(x : Integer) : Integer

'/'(x : Integer) : Integer

abs() : Integer

## **String**

=(s : String) : Boolean

<>(s : String) : Boolean

size() : Integer

concat(s : String) : String

toUpper() : String

to Lower() : String

substring(from : Integer, to : Integer) : String