

Algoritmus — folyamatábra — program

A Magyar Szocialista Munkáspárt Központi Bizottságának 1982. évi állásfoglalása többek között leszögezi, hogy „az iskolarendszer távlati fejlesztésében a fejlesztés legfontosabb elemének a közoktatás tartalmi korszerűsítését kell tekinteni”, s ezen belül „megkülönböztetett figyelmet kell fordítani a korszerű műveltség megalapozására, a képességek fejlesztésére és a folyamatos önművelés készségeinek kialakítására. Biztosítani kell, hogy az új, tudományos ismeretek és a kor követelte igények fokozatosan beépüljenek az iskola műveltségi anyagába.”

A számítástechnika és annak alkalmazása ilyen „új tudományos ismereteket” jelentenek, ezért a számítástechnikai oktatás közép- és hosszú távú célkitűzései megfogalmazzák a jelen öt éves tervben (ill. a továbbiakban) a közép- és általános iskolák számítástechnikai oktatási feladatait, a számítástechnika-alkalmazási ismeretek széles körű elterjesztését.

Örvendetesnek ítéljük meg, hogy a Művelődési Minisztérium – a fenti célkitűzések megvalósításáért – már az 1984/85. tanévtől az általános iskolai *fakultatív foglalkozások* keretében bevezethetőnek ítéli meg „A számítástechnika alapjai” című tárgyat, továbbá, hogy a Módszertani Közlemények hasábjain is jelenik meg – lelkes pedagógusaink tollából – a számítástechnikai kultúrát terjesztő módszertani cikk. Ezért is érezzük szükségét annak, hogy jelen írásban röviden szóljunk a számítógépes feladatmegoldás – elsősorban matematikai feladatmegoldás – elvi és oktatásmódszertani problémáiról. Ezen belül definiálunk néhány fontos számítástechnikai alafogalmat is (másokat csupán megemlítünk), de mindenekelőtt arra szeretnénk rámutatni, hogy mi mindent kell tennünk ahhoz, hogy egy feladatot számítógép segítségével oldhassunk meg.

A feladatmegoldás folyamatsorából most csak három fontosnak tartott lépést – algoritmus-keresés, folyamatábra-készítés, programírás – emelünk ki. Ugyanis ezeket elengedhetetlennek véljük (egyesteket számítógéptől függetlenül is bevezethetünk iskolai oktatásunkban), olyan oktatási alapelvek megvalósításához, mint pl. az algoritmikus gondolkodás fejlesztése, az átgondolt tervszerű munka végeztetése, a konstruktív gondolkodásra nevelés stb. Mondanivalónk szemléltetéséhez mindössze egyetlen feladatot tűzünk ki.

Algoritmus

Tekintsük a következő feladatot:

számítsuk ki az $y = 3x - 5$ függvény helyettesítési értékeit a $[0, 10]$ zárt intervallumon 0,2 lépésekkel. A számítás eredményeit foglaljuk táblázatba!

Írjuk le először a feladat megoldásához szükséges lépéseket.

1. lépés: x első értékének megadása (x értéke legyen egyenlő nullával).
2. lépés: Számítsuk ki y értékét az $y = 3x - 5$ hozzárendelés alapján.
3. lépés: Jegyezzük fel a táblázatba x és y értékét.
4. lépés: Legyen most x értéke 0,2-del több, mint eddig volt (x új értéke legyen $x + 0,2$).

5. lépés: Vizsgáljuk meg, hogy x értéke eleme-e a $[0, 10]$ intervallumnak, és ha igen, akkor ismételjük meg a fenti lépéseket a 2. lépéstől, míg ha nem, akkor
6. lépés: a számítás befejeződött. A feladatot megoldottuk.

A számítás ilyen előírásokkal való leírását, amelyekkel a kiindulási adatokból a számítás eredményeit létrehozó elemi lépéseket és azok sorrendjét is meghatározzuk, a feladat megoldási algoritmusának, eljárásának nevezzük. Vegyük észre, hogy a fenti algoritmus véges számú lépésben (nem véletlenszerűen) vezet el a kiindulási adatokból a számítás eredményeihez.

Az általánosítás kedvéért vizsgáljuk meg az algoritmus egyes lépéseit abból a szempontból, hogy milyen tevékenységre utasítanak. Így

- az 1., 2. és 4. lépés aritmetikai művelet(ek),
- az 5. lépés logikai művelet (döntés) elvégzésére,
- a 3. lépés az eredmények kiírására,
- a 6. lépés pedig a számítás befejezésére utasít.

Általában is az elemi lépéseket a következő csoportokba sorolhatjuk:

- adatbevitel,
- aritmetikai értékadás,
- logikai értékadás,
- adatkiírás,
- és szervezési lépések.

Végül megadjuk a számítógépes algoritmus fogalmát: szabályok egyértelműen meghatározott véges halmaza, amely szerint valamely feladat véges számú lépésben megoldható.

Megjegyezzük, hogy az algoritmus fogalom a most definiálnál sokkal általánosabb, általánosabbá tehető, de az általánosítási törekvések és eredmények tárgyalása nem célja az itt elmondandónak.

Folyamatábra

A feladat algoritmikus leírását szemléletessé tehetjük az algoritmus rajzával, folyamatábrájával.

A folyamatábra egyik nagy előnye – az általános információ-továbbítás szempontjából, így az algoritmikus szemlélet kialakítása, a számítástechnikai elemek oktatása szempontjából is – hogy számítógéptől független, így „öncélúan” is használható. Ha tanítványainkat csupán arra szoktatjuk rá, hogy a feladat megoldási lépéseit folyamatábrán szemléltessék, akkor már nagy lépést tettünk előre a számítástechnikai elemek oktatásának területén. A folyamatábra másik nagy előnye pedig éppen az, hogy összekötő kapcsolatot jelent a feladatmegoldás algoritmusára és ennek számítógépes programjának megírása között.

A folyamatábra egy feladat meghatározásának, elemzésének vagy megoldási módjának olyan grafikus ábrázolása, amelyben az adatokat, a műveleteket, a kapcsolatokat, az egyes lépéseket jelképek ábrázolják. A legegyszerűbb (és szabványosan is használt) jelképek: az irányított folyamatvonal, a be- vagy kimenettel ellátott kör, az egyetlen kimenettel ellátott téglalap és a két kimenetet tartalmazó rom-

busz. Az algoritmikus lépések végrehajtási sorrendjét az irányított folyamatvonal határozza meg.

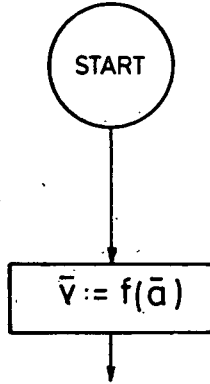
Minden folyamatábra négy folyamatábra-elem ismétlésével állítható elő.

A négy folyamatábra-elem ismertetéséhez először vezessük be a következő jelöléseket:

1. az input változók vektorát: $\bar{a} = (a_1, a_2, \dots, a_i)$, amely a megadott adatokból áll, és így a számítás alatt nem változik (adatvektor);
2. a programváltozók vektorát: $\bar{v} = (v_1, v_2, \dots, v_j)$, amely a számításhoz szükséges változókból áll, és így értékeket a számítás során kap (a változók);
3. az output változók vektorát: $\bar{e} = (e_1, e_2, \dots, e_k)$, amely a számítás befejezésekor az eredményeket adja (eredményvektor).

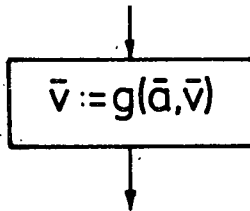
E vektorok bevezetésével egyidejűleg három (nem üres) tartományt is definiáltunk: az input tartományt (jelölje: T_a), a programtartományt (T_v) és az output tartományt (T_e). Ezek után a négy folyamatábra-elemet a következőképpen adhatjuk meg:

1. kezdőelem:



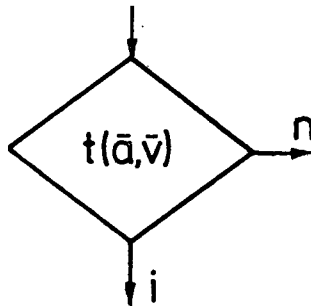
ahol f egy a T_a input tartományt a T_v programtartományba leképező függvény.

2. értékadó-elem:



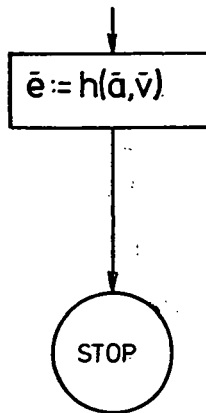
ahol g egy a $T_a \times T_v$ -t (a tartományok Descartes-szorzatával előállt tartományt) T_v tartományba leképező függvény.

3. logikai elem:



ahol t egy a $T_a \times T_v$ tartomány feletti reláció.

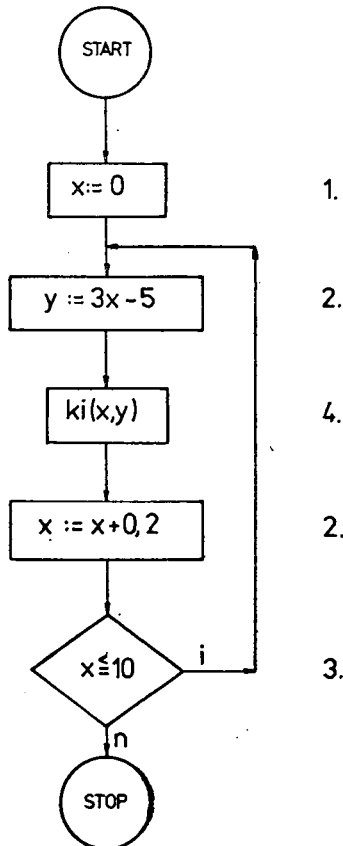
4. vég-elem:



ahol h egy a $T_a \times T_v$ tartományt a T_e output tartományba leképező függvény.

Ezek után megjegyezzük, hogy a négy elem ismétlése úgy értendő, hogy a folyamatvonal egyetlen kezdő- és végelemet tartalmazhat, megengedett két vagy több folyamatvonal csatlakoztatása, és nem tartalmazhat „zsákutcát” jelentő logikai elemet, az adott függvények és relációk pedig az adott tartományokon értelmezettek.

Kitűzött feladatunk megoldási algoritmusát az elmondottak alapján szemléltetjük a következő folyamatábrával:



ahol az ábra mellé írt számok (1, 2, 4, 2, 3) az egyes folyamatábra-elemek sorszá-
mát jelölik, továbbá

$$\bar{a} = (0, 10, 0, 2, 3, 5),$$

$$\bar{v} = (x, y),$$

$$\text{és } \bar{e} = (x_1, y_1, x_2, y_2, \dots, x_n, y_n)$$

az input-, program- és output változók vektorát és

f : az aritmetikai egyenlőség,

g : az aritmetikai összeadás, kivonás és szorzás,

t : a kisebb vagy egyenlő reláció,

h : a kiírás

műveleteit jelentik.

Rámutatunk feladatunkkal kapcsolatosan arra is, hogy a logikai értékadásnál lét-
rejött elágaztatást visszacsatolással oldottuk meg, s így folyamatábránk példa a szá-
mítógépes feladatmegoldás során sokszor alkalmazható ciklus szervezésére is. (Cik-
lusnak nevezzük a többször ismétlődő műveletekből, tevékenységekből álló utasítás-
csoportokat.)

Program

Most már következhet a számítás algoritmusának – a folyamatábra sémáinak
figyelembevételével – a számítógép nyelvére történő lefordítása, a számítógépi prog-
ram elkészítése. A számítógépi program olyan terv vagy ütemterv, amely meghatá-
rozza – a számítógép által végrehajtható alakban – a feltétlenül vagy feltételesen
elvégzendő tevékenységeket.

A program elkészítéséhez ismernünk kell a rendelkezésünkre álló számítógép
„nyelvét”, azaz azokat a parancsokat, utasításokat, melyeket számítógépünknek ad-
hatunk, és azokat a szabályokat, melyek szerint az utasításokat, az utasítások egy-
másutánjait írunk kell, illetve írunk szabad.

Tételezzük fel először, hogy egy PTK 1050 zsebszámológépünk van. Ekkor fel-
adatunk megoldási programja a következő lehet:

00	STO 0	1.
01	2nd Pause	
02	x	
03	3	
04	–	
05	5	
06	–	2.
07	2nd Pause	4
08	RCL 0	
09	+	
10	·	
11	2	
12	=	2.
13	INV 2nd x \cong t	3.
14	RST	
15	R/S	
<hr/>		
10	STO 7	} adatbevétel
0	STO 0	
<hr/>		

Megjegyezzük, hogy a szünetek beiktatása az értékpárok lejegyezhetősége végett szükséges, és a jobboldali sorszámok a folyamatábra megfelelő sorszámú blokkjaira utalnak.

Másodszor álljon rendelkezésünkre olyan számítógép (pl. ABC-80, HT-1080Z, WANG stb.), amely – egy magasabb szintű nyelven – a BASIC programozási nyelven programozható, ekkor feladatunk megoldási programja a következő lehet:

```
10 X = 0                1.
20 Y = 3*X - 5         2.
30 PRINT X, Y          4.
40 X = X + .2          2.
50 IF X <= 10 THEN 20 3.
60 STOP
```

ahol most is a jobboldali sorszámok a folyamatábra megfelelő blokkjaira utalnak.

Ha felhasználjuk, hogy a BASIC nyelvnek ciklusszervező utasítása is van, akkor a program így is írható:

```
10 FOR X = 0 TO 10 STEP .2
20 PRINT X, 3* X - 5
30 NEXT X
40 STOP
```

ahol rövidebben utasítottuk a számítógépet ugyanazon tevékenység végrehajtására.

Kitűzött feladatunk így számítógépünkön lefuttatható. Bár most az egyes tevékenységeket – algoritmus-keresés, folyamatábra-készítés, programtervezés és programírás – e cikkben készen kaptuk, mégis e rövid ismertetésből is látható, hogy a számítógépes feladatmegoldás komoly szellemi tevékenység, mely pontos, igényes, állandó ellenőrzés melletti folyamatokból tevődik össze. Elengedhetetlen azonban a matematika korszerű oktatásában – e területen is – a bátor kezdeményezés, hogy a felnövekvő nemzedék önállóan gondolkodó, kezdeményező és alkotó személyiséggé váljon.

IRODALOM

- [1] Általános iskolai fakultatív foglalkozás: A számítástechnika alapjai. OPI Útmutató, 1983.
- [2] Így tanítjuk a matematikát 2. kötet. Tankönyvkiadó, Budapest, 1978.
- [3] Magyar Népköztársasági Országos Szabvány: MSZ 7788, 1981.
- [4] MSZMP KB 1982. április 7-i állásfoglalása. Kossuth Könyvkiadó, Budapest, 1982.
- [5] *Perge-Puskás*: Numerikus és gépi módszerek 1. kötet. Tankönyvkiadó, Budapest, 1974.
- [6] *Puskás*: BASIC programozási nyelv. JGY Tanárképző Főiskola Tudományos Közleményei, Szeged, 1977.
- [7] *Z. Manna*: Programozáselmélet. Műszaki Könyvkiadó, Budapest, 1981.

