

Hogyan tanítunk meg az orosz igék parancsoló módjának képzésére egy embert és hogyan — egy elektronikus számológépet?

1. Bizonyos terminológiai kérdésekkel és általánosságokkal kell kezdenem.

Az embert tanítjuk, az állatot idomítjuk (olykor erre is azt mondjuk: tanítjuk), a gépet programozzuk. Bizonyos szempontból a tanítás, az idomítás, a programozás egy kalap alá vonható: e műveletek elvégzésének eredményeképpen valakit vagy valamit képessé teszünk olyan feladatok önálló (tehát az oktató, az idomító, a programozó esetenkénti beavatkozása nélküli) elvégzésére, amire ő (az) korábban nem volt képes és önmagától feltehetően nem is lett volna képes. Ezért az alábbiakban a rövidség kedvéért a *tanítás* összefoglaló kifejezéssel fogok élni olykor, amikor a tanítás és a programozás bizonyos közös vonásairól szólok. (Jobb volna természetesen, ha a három folyamatnak volna egy külön összefoglaló elnevezése.) — Meg kell jegyezni még a következő terminológiatörténeti tényt. Maga a *programozni* (*programozás, programozott* stb.) kifejezés-család a didaktikában, tehát az emberi tanítás tudományában vetődött fel, jó két évtizeddel azelőtt, hogy a mai programozható gépekről egyáltalán beszélhettünk volna. A programozott oktatás a legutóbbi években világszerte újból közkeletű lett. Semmiféle adat nem áll e téren a rendelkezéseimre, de lehetségesnek tartom, hogy e fellendülés egyik oka a gépi programozás terén nyert tapasztalatoknak az emberi tanításban való visszaalkalmazhatósága. Mindenesetre az alábbiak bizonyos fényt fognak vetni a programozott oktatás kérdéskörére is, a programozott oktatás és a gépi programozás kapcsolatára.

A következő kérdés már tartalmi jellegű. Ahhoz, hogy valamit meg tudjunk tanítani, két dologra van szükség: *a)* jól kell ismernünk a megtanítandó anyagot; *b)* bizonyos módszert kell követnünk az anyag elsajátíttatása folyamán. Természetesen szükségünk van egy olyan alanyra is, aki (amely) a tanításra — idomításra — programozásra alkalmas, ezt azonban magától értetődőnek tekintem. Az első feltétel triviálisnak tetszhet: csak egyes hősi időszakokban, vígjátékokban és nem nagy számú egyéb helyzetben fordulhat elő, hogy olyanok tanítanak, akik maguk is csak kevéssé ismerik az elsajátítandó anyagot. Különbőség lehet persze magában az elsajátítandó anyagban, így annak ismerete is lehet az ember számára könnyebben vagy nehezebben áttekinthető: az idomítás során viszonylag (számunkra) igen egyszerű anyagot sajátíttatunk el (például: kifli gyúrása majommal, háromig való számoltatás elefánttal stb.), míg a tanítás és a programozás során felvetődhetnek számunkra igen egyszerű feladatok (például: írás-olvasás), de egészen bonyolultak is (például: differenciálszámítás, parancsoló mód képzése nem anyanyelvben stb.). Ami a második feltételt, a módszert illeti, az egyaránt fontos a tanítás mindhárom válfajában; sőt talán egyaránt nehéz is. Szempontunkból igen lényeges megjegyezni, hogy többféle módszer lehetséges egy és ugyanazon anyag elsajátíttatása esetében is.

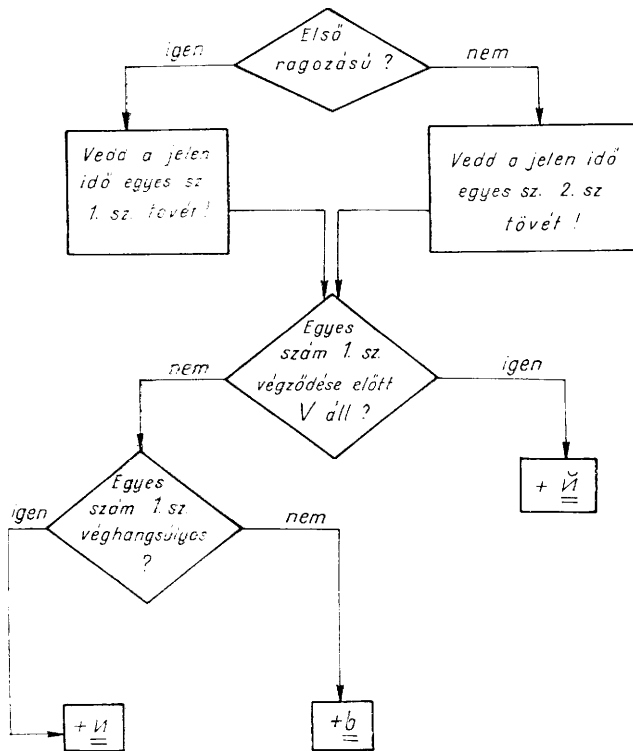
Ez a módszerbeli többféleség az állatok esetében a veréstől a kedvelt csemege osztásáig mindenféle lehet — az emberi oktatásban, konkrétan a nyelvoktatásban sokat hallunk direkt, grammatizáló stb. módszerről. A gép esetében is többféle módszer volna lehetséges. Miaz alábbiakban annak a kísérletünknek az eredményeire támaszkodunk, amelyet a KLTE ODRA 1013-as számú (lengyel gyártmányú) gépén végeztünk. Ez a gép programvezérlésű, vagyis valamely feladat elvégzéséhez szigorú, lépésről lépésre minden teendőt megadó, a gépet minden belátható eshetőségre felkészítő programot kell adnunk neki ahhoz, hogy kitűzött feladatát el tudja végezni, önállóan, a programozó beavatkozása nélkül. Könnyen belátható a feladat ilyen megfogalmazásának rokonsága a programozott oktatás célkitűzésével. Hiszen ez utóbbi éppen abban különbözik a „nem programozott” oktatástól, hogy lépésről lépésre vezeti a tanulót az anyag elsajátításában; hogy rávezeti azokra a logikai lépésekre, amelyekre a tanulónak valamely feladat megoldása során szüksége lesz. A programozott oktatás tehát (talán ki sem mondva) az embert programvezérlésű automatához hasonlónak tartja bizonyos vonatkozásban — lehet, hogy az ember az, lehet, hogy nem; erre most egyáltalán nem kívánok kitérni. A középiskolás matematika-tankönyvek, úgy látom, régóta közlik az egyes feladatok helyes megoldását: itt tehát nem maga az eredmény a fontos, hanem az az út, ahogy az adott eredményhez a tanuló eljut; vagy pontosabban szólva: persze lényeges az, hogy a tanuló helyes végeredményre jusson, de a tanárnak elsősorban nem ezt kell ellenőriznie, hanem azt, hogy milyen úton jutott a diák ehhez az eredményhez. Más tárgyakban, így a nyelvoktatásban, nyilván korábban is fontos volt a megtett logikai út is, de azért főleg vagy csakis a végeredményt ellenőrizték. Azt hiszem, a programozott oktatás az első olyan módszertani eljárás-komplexus, amely — a matematika korábbi gyakorlatához hasonlóan — a megtett utat központba állítja; amely aktívan be kíván avatkozni abba is, hogy hogyan, milyen módon jusson el a tanuló a kívánt eredményhez. Ennek megfelelően közli is a tanulóval ezt az utat, a megoldás logikai menetét elsajátíttatja a tanulóval. Nos, a programozó sem tesz mást gépe esetében, mint a legutóbb említettet: elsajátíttatja vele a megoldás logikai menetét. Hangsúlyozni szeretném, hogy érzésem szerint ez a célkitűzés az ember esetében sem új, csupán annak központba állítása az. Így például amikor a latintanulás logikai készségeket fejlesztő, gondolkodást fejlesztő voltáról szövegünk és szólunk, nyilván valami ilyenféle lebegett a szemünk előtt: lehet, hogy a tanulónak magára a kész eredményre (ti. az egyes latin alakokra, mondatokra stb.) soha nem is lesz szüksége, de egyáltalán nem haszon nélkül való számára maga az az út, melyet ezen eredmények elérése érdekében bejárt!

2. Milyen logikai lépések kellene ahhoz, hogy az ember helyesen tudja az egyes orosz igék parancsoló módját képezni? Ezzel a kérdéssel részletesen foglalkozom nemrég megjelent egyetemi orosz leíró nyelvtanunkban (*Курс современного русского языка*. Tankönyvkiadó, Budapest, 1968. 384 kk.). Ezért engedtessek itt meg nekem, hogy a logikai menetet egyetlen ábrán érzékeltessem és csupán a szabályos (nem rendhagyó) alakokra korlátozzam magam (l. l. ábra).

Magát az ilyen ábrát blokkdiagramnak hívjuk, említett könyvünkben az itt vizsgált kérdéssel kapcsolatosan is, de másokkal kapcsolatosan is nem egy ilyet találhat az olvasó. A blokkdiagram megértéséhez csupán annyit kell tudnunk, hogy kétfajta elemet tartalmaz: mindig csak „igen”-nel vagy „nem”-mel megválaszolható kérdéseket (ezeket rombuszokba szoktuk beleírni — ez természetesen nem lényeges, csak szokás) és utasításokat (ezeket meg téglalapokba szoktuk helyezni). E kétfajta elemeket nyilak kötik össze egymással: a nyilak mentén kell haladnunk, azok irányában. (Egy szép és az itteninél jóval érdekesebb blokkdiagramot találhat az olvasó például Mel'uk cikkében, mely a magyar főnévragozásról szól: MNy. 64. 189.) Könnyen belátható, hogy

az ilyen sémák, bár természetesen nem tartoznak a lényeghez, igen alkalmasak arra, hogy rajtuk bizonyos feladatok megoldásának logikai menetét érzékeltesük. Próbáljuk ki az általunk közölt blokkdiagramot három példán.

a) **читáть**: Első ragozású ige, tehát a tőnek azt a változatát kell választanunk, amely az egyes sz. 1. személy végződése előtt szerepel (ez itt még nem lényeges): *чита-*. Ez a tőalak magánhangzót jelölő betűre (ti. az *a*-ra) végződik, tehát a *ѣ* végződést kell



1. ábra. V = magánhangzót jelölő betű (beleértve a „w” jelet, a hangsúly jelét is)

hozzáilleszteniünk: **читай**. (Az embernek – sapienti – ennyi elég is. Korrekt gépi program esetében ezt követően – és minden végeredményt követően – még legalább egy utasítást kellene adnunk: „Írd ki az így kapott eredményt”, mert enélkül a gép azt megtartaná valahol a memóriájában, számunkra hozzáférhetetlen és élvezhetetlen módon. Nem mondom, olykor némely ember tanulók is viselkednek így, kijelentvén: „En ezt tudtam, tanár úr, csak nem tetszett mondani, hogy mondjam [írjam fel a táblára stb.]”.)

b) **сказáть**: Első ragozású ige, tehát az egyes sz. 1. személyű végződés előtti tőváltozatot kell vennünk: *скаж-* (ti. az alapul vett alak így van: *скажѹ*). Mivel e tőváltozat nem magánhangzót jelölő betűre végződik, még egy ellenőrzést kell tennünk: Véghangsúlyos-e az egyes sz. 1. személyű alak? Igen, véghangsúlyos, tehát a végződés: *ѹ*, az eredmény: **скажи**.

c) **броси́ть**: Második ragozású ige, tehát a második személyű végződés előtti tőváltozatot kell vennünk: *брос-* (az első személyű így lenne: *брошу*, *ш*-vel). Mivel e tővál-

tozat nem magánhangzót jelölő betűre végződik, ellenőrzendő: Véghangsúlyos-e az egyes sz. 1. személyű alak? Nem, nem az, tehát a végződés: *ь*, az eredmény: **брочь**.

És így tovább — az olvasó ezer és ezer orosz igeen kipróbálhatja kis automatánkat: az minden nem rendhagyó esetben helyes eredményre fogja vezetni. Ha úgy tetszik, készíthet e rajz alapján egy kis oktató célgépet: a nyilak helyén legyenek drótok, a szövegek (kérdések és utasítások) alá tegyünk egy-egy zseblámpaégőt, a kérdések alá ezenkívül egy-egy háromállású kapcsolót. Ha e kapcsolókat rendre a megfelelő irányba elfordítjuk, mindig a helyes következő lépés lámpáját gyújtottuk ki, legyen az kérdés vagy utasítás. Az égő lámpák egy-egy ige után mutatni is fogják, milyen utat jártunk be. (Az embernek ennyi megint elég. Gép esetében még egy utasítást kellene most adnom: „Miután így örömmel végigszemlélted megtett utad lámpáskáit, oltsd el őket mind [mert különben a következő ige esetében összezavarodsz, vagy, ha nem játszol tovább, zseblámpaelemed ki fog merülni]”!))

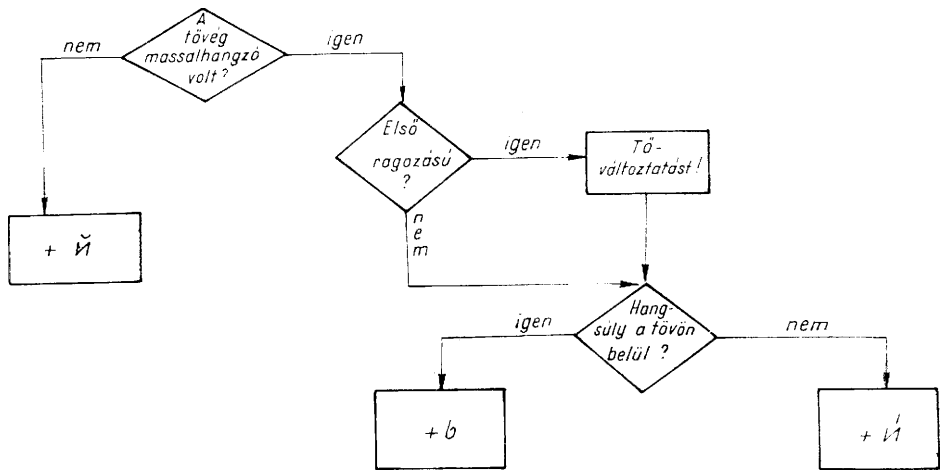
Amikor tehát az ember megkapta egy-egy ige infinitivusi alakját, akkor erről az igeről még a következő adatokat kellett ismernie: *a*) az egyes első személy, hangsúllyal együtt; *b*) az egyes második személy; *c*) első ragozású-e az ige; ezenkívül *d*) meg kellett tudnia különböztetni a magánhangzót jelölő betűket a mássalhangzót jelölőktől. Az *a*)–*d*) adatok ismeretében, valamint annak ismeretében, hogy a parancsoló mód végződése vagy *ь* vagy *у* vagy *ь*, minden egyes nem rendhagyó ige esetében helyesen ki tudjuk alakítani a parancsoló mód tövét és ehhez ki tudjuk választani a megfelelő végződést az 1. ábrán sematikusán vázolt algoritmus alapján. Már az ember esetében is gondolni lehetett arra (az oktatás gyakorlatában így is szoktunk tenni — bár ez nem minden orosz tanárra érvényes), hogy valamivel gazdaságosabban tároljuk ugyanezeket az információkat. Az *a*) és *b*) információk nagyobb része összesűrithető a következő módon: Az infinitivusi alakban függőleges vonallal jelöljük a *т* végét, így: *чумá/мь*, *каз/ámь*, *броч/умь* — ebből a leírás módból automatikusan kiszámítható, milyen lesz a *т* az egyes első, illetve az egyes második személyben. A fennmaradó hangsúlyproblémát egy további szabállyal lehet megoldani: e szabály ismeretében az infinitivusi alak hangsúlyából kiszámítható az egyes első személy hangsúlya. (Szerencse, hogy csak erre van szükség. Már az egyes második hangsúlya az infinitivus hangsúlya alapján nem látható előre.) Vagyis a fenti *a*)–*b*) helyett ez írható első ismerendő adatként: *a*) a tövég és a hangsúlynak a helye az infinitivusban. E módosítás elfogadásával természetesen bővítenünk kell az eredeti algoritmust, körülbelül így: A tövég mássalhangzó volt? Ha igen, első ragozású ige volt? Ha igen, akkor a tövéget megadott módon meg kell változtatni (a mi példánkban: a *з*-ből *ж*-t kellett csinálnunk), ha nem — a tövéget változatlanul kell hagynunk. Továbbá be kell vezetnünk egy lépéscsoportot a hangsúly ellenőrzésére, körülbelül így: A tövön belül volt a hangsúly? Ha igen, a korábban kialakított *т*höz *ь*, ha nem — *у* illesztendő végződésként. Algoritmusunknak ezt a módosított változatát közli a 2. ábrán feltüntetett blokkdiagram. (Pusztán esetleges orosz tanár olvasóim kedvéért jegyzem meg, hogy itt a „Tövet változtass!” utasításba van beürítve nemcsak az ismert *з ~ ж*, *с ~ ш* stb. váltakozás, hanem az infinitivusban *оамь* végű igék *у*-ra változtatása is: *пуч/оамь ~ пучь*-. Erre a sűrítési lehetőségre Jánoska Sándor kollégám hívta fel a figyelmemet.)

Ha egyes gyakoribb kivételeket ki akarunk szűrni, akkor természetesen még tovább kell bonyolítanunk algoritmusunkat; az egészen egyedi kivételeket nyilván nem érdemes az algoritmusba sűríteni, azokat célszerűbb lexikálisan elsajátítani.

3. Az a sűrítés, amely az 1. ábrától a 2.-hoz vezetett bennünket, egyes olvasók számára céltalannak tűnhetett. A valóságban, mint említettük, számos orosz tanár már eddig is élt ezzel a módszerrel, vagyis a *т* függőleges vonallal való leválasztásával az

infinitivusban, mely nem rendhagyó igék esetében a teljesebb averbót a fent jelzett módon helyettesíthette. Itt tehát csak azt mondhatom olvasóimnak — higgyük el orosz tanár kollégáinknak, akiknek ez a napi gyakorlatuk, hogy ez a módszer legalábbis nem rossz már az ember esetében sem.

A gép esetében pedig, melyről ebben a pontban kívánnék szólni, egy ilyen tömörítés nagyon szükségesnek mutatkozott. Természetesen semmi akadályja sem lett volna annak, hogy minden egyes infinitivus után közöljem a géppel az egyes szám első és másó-



2. ábra. Megjegyzés: Az első kérdést így is fel lehetett volna tenni: „Tővég V?” (= magánhangzót jelölő betű); az „igen” és „nem” válaszok ebben az esetben természetesen helyet cserélnének

dik személyű alakot is: ahogy az infinitivust beolvasta és a szükséges mértékben megértette, éppúgy persze ezeket a segédalakokat is feldolgozta volna. De minek, ha ezek az alakok egyetlen függőleges vonallal és néhány szabállyal pótolhatók? A mai gépeknek gyakorlatilag korlátlan nagy a memóriájuk — abba természetesen belefér akárhány kiinduló igealak. A mai gépek néhány tíz- vagy néhány százezer műveletet végeznek el egyetlen másodperc alatt — tehát másrésről az sem probléma esetükben, ha néhány lépéssel bonyolódik az algoritmus — néhány ezred vagy százezred másodpereni idővel többet fognak gondolkodni egy-egy igealak képzésekor. (Az általunk alkalmazott ODRA gép ugyan kivétel volt mindkét tekintetben: csak 600 — 1200 műveletet végez el másodpercenként és mintegy másfél nyomtatott ívnyi szöveggel „dugig” megtölthető egész memóriája — ám a jelen feladat megoldása során e lényeges korlátai nem voltak észrevehetőek. Az ember számára a „néhány ezred” vagy „néhány század” másodperc egyaránt kevésbé érzékelhető.) Tehát nyugodtan választhattunk a két lehetőség között: közöljük-e egy teljesebb averbót, és ahhoz viszonylag egyszerűbb szabályokat fűzzünk; vagy közöljük csak az infinitivust (a rajta bejegyzett tővéggel) meg az 1. ragozáshoz való tartozást (nem tartozást) és egy talán bonyolultabb algoritmust. Mi ezt az utóbbi utat választottuk, és egészen biztosak vagyunk benne, hogy minden más programozó is ezt tette volna: igyekezett volna megtalálni a lehető legkevesebb bemenő információt, amely azonban még mindig elegáns szabályrendszerrel kezelhető. Nem tudnám pontosan megmondani, miért, de a gép ösztönzi a vele dolgozó embert a maximális gazdaságosságra.

Emberekkel dolgozva, azokat tanítva akár ugyanarra, sohasem éreztem ezt a kényszert: épp ellenkezőleg, előadásaim során, azt hiszem, igyekszem minél vidámabban s ezért eléggé lazán tálnai az anyagot, azt körüljárni minden oldalról, megvilágítani példákkal, rajzokkal, megtűzdelni hasonlatokkal és — már ahol lehet — tréfákkal. (Igaz, hogy minden egyes ilyen „laza körüljárás” magvát ott is egy-egy maximális gazdaságosságra törekvő építmény teszi — az megint más kérdés, hogy sokszor nem találok meg a legjobb megoldást, és örömmel veszem át valamelyik kollégám tömör szabályát, elegáns anyagrendezését.)

Ez a tönörítésre törekvés meg a mi gépünk sajátosságainak ismerete együttesen szülte egyébként a következő ötletet — valamilyen formában talán hasznos lehet ismét az emberek oktatásában is. Kezdetben a gépet úgy próbáltam volna tanítani, ahogy idáig az emberekkel tettem: a bejelölt tövég után nézze meg, *ovaty* ~ *ovawty* vég áll-e utána; a tövég utánra esik-e a hangsúly stb. (Megjegyzendő: a továbbiakban az élethűség kedvéért úgy írom le az orosz szavakat, ahogy azokat a csak latin betűket ismerő ODRÁ-val közöltem, ahogy ő azokat nekem vissza leírta. Ebben az átírásban az orosz lágyjelnek — a magyar analógiát tartva szem előtt — a latin „y” felel meg, a hangsúlyt a hangsúlyos magánhangzó után tett „w” betűvel jelöltem — erre a betűre ui. úgy sincs más szükség oroszról való átírás esetén —, egyebekben pedig az angol alapú telex átírást követtem. Eszerint orosz „e” — átírva mint „s”, orosz „u”, „ж” átírva mint „sh”, „zs” stb. Természetesen van latin és cirillbetűket egyaránt ismerő gép is, ilyen például a Budapesti Egyetemi Számítógéppont Razdan nevű gépe.) Nagy, apai íróasztalomnak szinte a felét betöltő blokkdiagram lett ebből — ez, mint fentebb említettem, eleve nem tetszik a programozónak. Amint azután forgatni, rakogatni kezdtem az egyes, tövég-jellel ellátott infinitivusokat, a következőre jöttem rá (a telexet ismerő olvasóim látni fogják, hogy némi egyszerűsítéssel közlöm a dolgot, de a lényegen ez semmit sem változtat):

(moch/y 1	моч/ь 1)
duwma/ty 1	ду'ма/ть 1
(viwj/ti 1	вы'й/ти 1)
brows/ity 2	бро'с/ить 2
(nes/tiw 1	нес'ти' 1)
skaz/awty 1	сказ'а'ть 1
besewd/ovaty 1	бесе'д/овать 1
ris/ovawty 1	рис/ова'ть 1

(A cirillbetűs oszlopot csak itt közlöm, a könnyebb azonosíthatóság végett.) Vagyis: ha a tövéget jelölő törtvonás után egyet olvasunk és ott *y*-t találunk, akkor egy *moch/y*-féle kivétellel van dolgunk; ha ott nem *y*-t találtunk, tovább olvasunk eggyel: ha ekkor találunk *y*-t, ez általában egy *chitaw/ty*, *duwma/ty*-féle, tehát *j* ragot felvevő ige lesz (hangsúlyval a tövön), ha viszont ekkor *i*-t találunk, akkor ez egy *viwj/ti*-féle kivétel lesz; ha nem *y* vagy *i* volt ezen a helyen, még eggyel tovább olvasunk s akkor, ha *y*-t találunk, ez egy *brows/ity*-féle ige; ha *w*-t: *nes/tiw*-féle — és így tovább, a gép számára készített blokkdiagramomnak ezt a részét a 3. ábrán közlöm. Ha a kivételeket (melyek a fenti felsorolásban zárójelek között állnak) nem tekintem, akkor a következőt mondhatom: Számoljuk meg a tövég után álló betűket. Ha még két jel következett ez után, a rag *j* lesz; ha három — mássalhangzó után *y*, magánhangzó után *j*; ha négy — hangsúlyos *i*; ha öt — *uj*; ha hat — *uwj* (vagyis hangsúlyos *uj*, l. a fenti példát). Vagyis: a tövéig egyszerűen csak másoltatnom kellett; az ezt jelölő törtvonás után pedig — számoltatni és a számolás eredményének megfelelően az eddig kapott tőhöz (a ragozási típusnak megfelelően változtatotthoz vagy változatlanhoz) a lehetséges végződéseket valamelyikét

stb.)”, „Magánhangzót jelölő-e a beolvasott betű?”, „Írj ki valamit” (hogy éppen mi kellett kiírnia, azt, illetve annak a címét, természetesen az alapprogram határozza meg esetenként) és így tovább. Egy-egy ilyen általánosabb érvényű, máshol is felhasználható utasításcsoportot *szubrutin*-nak hívjuk. Az „Írj ki valamit” annyira létfontosságú utasítás, hogy az szerepel természetesen a gép „alapvető készségei” között, vagyis a gépet már a gyártó vállalat úgy szállítja, hogy memóriájának bizonyos részében fixen szerepel (illetőleg oda fixen bevihető) egy ilyen utasításcsoport, annak csak a címét kell megadni egy kb. ilyen tartalmú utasítással: „Hajtsd végre azt, ami az adott címtől kezdve memóriádban szerepel!” Nekünk mégis saját szubrutint kellett erre is fabrikálnunk különféle, itt nem részletezhető okokból. (Az ok nem az volt, hogy mi betűket írtunk ki és nem számokat: az „alapvető készségek” között szerepel egy szubrutin: „Számok kiírása” és egy másik: „Szöveg kiírása”.) A magánhangzóra nem csupán azért kellett a gépnek ráismernie, mert ilyen jel után mindig *j* (és sohasem *y*) áll. A *chítawjty*, *duwmajtj*-féle „rendes” igék közé, a pusztá számlálgatás után, bekeveredhettek *scsjty*, *grizjty* stb. -féle rendhagyók is, hiszen ez utóbbiak esetében is csupán két betű állt a tövéget jelölő törvonás után: ám ezeket könnyen ki lehetett szűrni úgy, hogy megállapítottuk: esetükben a *ty* előtt (tehát a tövégen) nem magánhangzó áll! Ilyen helyzet csak az itt két példával illusztrált kivételek csoportjában állhat elő. Másrészt az is világos, hogy a magánhangzójelek megkülönböztetése a mássalhangzókétól bármiféle nyelvészeti munka során majdnem ugyanolyan létfontosságú, mint az „Írj ki valamit” szubrutin. Ezért volt érdemes ezt is kiemelni az alapprogramból; ezért nem lenne fair eljárás a géppel szemben, ha az ide vonatkozó utasítások számát is hozzáadtam volna azokhoz az utasításokhoz, melyek valóban magának a parancsoló módnak a képeztetéséhez szükségesek a gépnek. (A túlságosan nagyszámú utasítás ui. a gép butaságát tanúsítaná – és/vagy a programozó járatlanságát is persze.)

Megjegyzem még azt, hogy az egyes gépi utasításoknak a fentebb jelzett alakja jellemző az olyan konstrukciójú gépekre, mint a mi ODRÁ-nk. Bármilyen felépítésű legyen is azonban valamelyik, az egyes utasítások (némi további finomságok mellett) valamilyen formában tartalmazza a következőket: *a*) mit kell tenni (összeadni, kivonni, összehasonlítani stb.); *b*) mivel kell ezt tenni (általában csak a címét adjuk meg annak a dolognak – számnak, betűnek stb. – amivel a teendő van); *c*) milyen címre kell menni ez után (milyen címen van a következő utasítás). Kiemelném a *c*) tényező fontosságát: ez biztosítja a gép folyamatos automatikus menetét, azt, hogy a gép fokról fokra valóban bejárja a szükséges logikai lépéseket. (Egy fentebbi példánkkal élve: ez gyújtja ki neki a következő lépés lámpását.)

4. A tanítás menetét, velejáróit, tempóját stb. számos olvasónk jól ismeri. Szeretném ebből a szempontból jellemezni a gép tanítását, a programozást.

A blokkdiagram készítése a tankönyv (nyelvtan) valamely fejezetének (paragrafusának stb.) a felvázolásához hasonlítható. Ennek megfelelően nehezen határozható meg a ráfordítandó idő – ugyanakkor ezt az időt még aligha szabad a tulajdonképpeni gépi tanuláshoz-tanításnak az idejébe beleszámítanunk. Nekem már évek, sőt évtizedek óta forog a fejemben az orosz parancsoló mód problematikája – persze, más problémák mellett. Számos jó leírásával is megismerkedtem az évek során e jelenségesopornak. (Itt jegyzem meg: Lotz János, Papp István, Jakab László és mások kutatásai, a MMNyR. leírása alapján, azt hiszem, szépen meg lehetne csinálni a magyar parancsoló mód programját is.) Tankönyvünk megfelelő részeinek az írásakor többé-kevésbé végleges formába is kellett öntenem ezt a dolgot – persze ezt megelőzően évről évre előadtam azt, sokszor egy évben többször is: nappal, este, levelezőknek, középiskolában. Így formálódott-érlelődött bennem ez sokáig. Magát a blokkdiagramot emlékezetem szerint több napon

át készítettem, első változatban, másodikban, sokadikban, természetesen megszakítá-
sokkal.

A második fázis még mindig nem a tulajdonképpeni tanításé (programozásé). A véglegesnek ítélt blokkdiagram alapján el kell készíteni a programot. Más szóval: az egyes rombuszok és téglalapok tartalmát valamilyen, a gép számára érthető nyelven le kell írni, az egyes utasítások és az utánuk következők címét megadni stb. Csak két példát erre: Az „Olvass egy betűt!” helyett mindenütt csupán ezt kellett leírnom:

026 00000 00000 00

(egyéb közölnivalóm pedig ezzel egyidejűleg nem volt a géppel, ezért lett a többi helyen nulla – ennyi és ilyen eloszlású nulla azért, mert konkrétan az ODRÁ-n dolgoztam). Még egy példa: Az „Írj ki valamit” helyett például ezt írhattam:

626 00000 17344 00

– itt a 17344 volt az a cím, amelyre konkrétan e kiíró utasítás után kellett mennie a gépnek. Kiírása válogatta, milyen címre kellett utána mennie. (Megjegyzem, ez a fentebb említett speciális kiíró szubrutinnak csupán egyik részutasítása volt.) Ez a fázis összesen nyolc-tíz, talán még annál is több órát vett el tőlem, szintén többszöri megszakítással – gyakorlott programozó számára jóval rövidebb lenne ez az idő.

E fázis két szükségszerű megszakítása az volt, hogy amikor programom elkészült, azt kipróbáltuk a gépen. (Hadd mondjak itt hálás köszönetet gép melletti matematikus segítőtársaimnak: Rochlitz Szilveszter és Jékel Pál tudományos mukatársnak.) Mind a két alkalommal három-négy órán át verejtékezve kerestük: hol s miért rossz a programom, mely persze nem akart helyesen működni. („Elsőre” hibátlanul működő programot talán még a legöregebb programozók sem láttak – igaz, nálunk a legöregebb programozó is harmadik x-ét tapossa, világviszonylatban talán a negyediket.) Így például első intrádára „csak” annyi volt a differencia, hogy az ODRA a *chítawjty* ige után kiírta: „kiveetel”, és ennek alapján nem volt hajlandó a parancsoló módját képezni; első ragozású igék esetében nem változtatta meg a tővégi mássalhangzót stb. – vagyis a külső szemlélő azt mondhatta volna, hogy éppenséggel semmit sem csinál. Pedig csinált persze, nem is keveset, de még rosszul.

Most lehet, hogy csalódást okozok az olvasónak. Jön ugyanis a tulajdonképpeni gépi tanulás ideje: a kipróbált, jó programot a gépnek be kell olvasnia, memóriájába kell vésnie stb. Ezt persze meg is teszi a gép, de ehhez még a mi, lassú kis gépünknek is összesen (tehát alapprogramostul-szubrutinostul) mindössze két-három pernyi időre van szüksége. Némi további programcsiszolás után fogunk adni a gépnek egy olyan utasítást, hogy öntse egy neki még megfelelőbb formába a kész programot. Ezt az utasítást a gép automatikusan végrehajtja (ellenőrzi is, hogy az új formába öntés hű volt-e az eredetihez és a „dobře” szó telexen való kiírásával jelzi a sikeres átírást). Az így, lyukszalagra (l. alább, 5.) kapott programot azután minden szükséges alkalommal tíz-tizenkét másodperc alatt be tudja vésni a memóriájába. A csalódást az okozhatja, hogy ezért a néhány másodpercért dolgoztunk idáig, hogy a címben feltüntetett „tanítás” tulajdonképpen csak ilyen rövid ideig tart. – Hadd tegyek ezzel kapcsolatban egy kitérőt. A földrajz és a csillagászat hozzászoktatta az embert ahhoz, hogy saját méreteinél sokkal sokkal nagyobb térbeli dolgokat elképzelhessen. A mikroszkóp bevezette az embert a nálánál sokkal-sokkal kisebb térbeli dolgok világába. Az idő viszonylatában a legutóbbi időkig csak az igen nagy intervallumokat kellett tudnunk elképzelni (geológia, csillagászat). Most mintegy idő-mikroszkópot kaptunk: bele kell élnünk magunkat a biológia felépítésünkhöz képest igen-igen kicsi időintervallumok világába.

És még egy pedagógus-észrevétel a gépi munka kapcsán. Miután a program elkészült, és a gép azt memóriájába teszi: egészen pontosan tudom róla, mit tud és mit nem tud. Tudom például, hogy a *viwnes/iti*, *viwjt/iti*-féle egészen ritka kivételeket felismeri, de már a gyakoribb *viwskaz/aty*-féle kivételt nem (ez utóbbiból a kívánt alakot így képezi: *viwskazhy*, lágyítójellel a végén, a helyes *viwskazhi* helyett). Ugyanígy *kriwknj*, *powmny* alakokat képez a helyes *kriwknj*, *powmni* helyett. (Megjegyzem, pillanatnyilag a program olyan, hogy a kivételekből n e m képezi a kívánt alakot, hanem a tő kifrása után megáll és ezt írja: „kiveetel, 1. kurs ... o., p.”, a pontok helyébe beírva fentebb idézett egyetemi leíró nyelvtanunk megfelelő lapszámait és paragrafusszámait. Főleg a magam lustasága miatt programoztam így a gépet — hosszá lett volna meg unalmas is a számomra az egyes kivételeket mind kiírom a saját könyvemből —; másrészt viszont abból a megfontolásból, hogy egy intelligens ember — és gép — nem tud okvetlenül mindent, de tudja, hogy hol kell utánanéznie.) Immár pontosan két évtizede programozok embereket (ha házitanítói működésemet is ideveszem, majdnem negyedszázada) — de ember tanítványaim esetében sohasem tudtam ilyen pontosan, mit tudnak és mit nem. A gép teljesen, kiszámítható pontossággal és unalmassággal az én gyermekem, az én agyam szerves kinövése — ezt eddig egyetlen ember gyermekemről sem mertem volna elmondani.

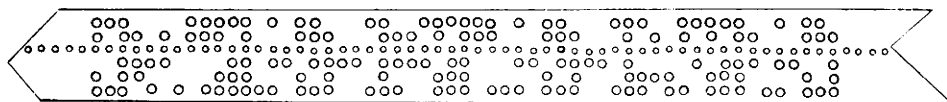
5. Milyen formában adjuk be a programot a gépnek, hogyan írja ki az eredményeket?

A programot — a mi gépünk esetében és számos más gép esetében — erős papírból készült szalagra lyukasztjuk egy közönséges telexgépen. Az ilyen gép szalaglyukasztás közben, mint rendes írógép, papírra, betűkkel is ír, mert így könnyebb ellenőrizni, nem követtünk-e el valamilyen hibát. A jó lyukszalagot visszük a gép beolvasó részéhez, amelyen az a fentebb jelzett idő alatt átfut — eközben természetesen semmi fizikai igénybevétele a szalagnak nincs, azt gyakorlatilag még akárhányszor fel lehet használni. (A lyukak helyén levő folytonossági hiányokat alakítja át a beolvasó áramimpulzusokká.)

Hasonló formában visszük be az infinitívusokat is, például így:

+chitaw/ty 1= +pis/awty 1= +brows/ity 2=

ez a szalagon pontosan úgy fest, ahogy azt a 4. ábra mutatja. A kiinduló (feldolgozandó) adatokat tartalmazó szalagot hívjuk adatszalgának: esetünkben az egyes igék előtt a „+” jel azért állt, mert a gép erről ismerte fel, hogy ott egy új ige kezdődik. (Megállapodhattunk volna persze másban is, például abban, hogy minden egyes igét új sorban kezdünk, akkor arról ismert volna rá.) Természetesen fontos, hogy előbb a program-szalagot adjuk be a gépnek — anélkül el sem tudja kezdeni az adatszalg olvasását.



4. ábra.

A továbbiakban a gép a program szerint dolgozik. Én úgy programoztam, hogy miután az adatszalg legelső jele (mondjuk, a *chitaw/ty* előtt álló „+” jel) beolvásodik, az olvasás leáll, és a gép a hozzá kapcsolt telexírógépen pontosan ezeket írja ki:

parancsoloomood keepzeese nem rendhagyoo orosz igeek infinitivusa
alapjaan

(a sor végén természetesen megáll és akárcsak egy gépirónó, visszatolja az írógéphengert

és sort emel). Majd új sorban (példánkban maradván) betűről betűre, egy jó gépfíró gyorsaságával, leír ennyit:

chitaw

Itt tulajdonképpen egy kicsit gondolkodnék, ám ez megint olyan kis idő alatt megy végbe, hogy nem tudjuk érzékelni, tehát ugyanolyan ütemben, mint eddig, kiírja a végződést a tő után és így megkapjuk:

chitawj

Ezt követően új sorra tér át, és egyenletes, gyors ütemben írja:

pishiw

Majd ismét új sor:

browsy

Ha adatszalogunknak (mint példánkban) itt vége volt, akkor a gép leáll. Újabb adatszalog esetében hasonló módon újra kezd dolgozni (de akkor már nem írja ki a címet: „parancsoloomood keepzeese stb.”). Mind a legelső esetben, mind e további adatszalogváltások után csupán annyit kell tennünk, hogy a gép „START” feliratú gombját megnyomjuk (megállni ő maga is megáll, de csak szalogvégen). Egy konkrét példa még a kivételek kezelésére. A bevitt adat a *nesjtiv* volt, erre ő ezt írta:

nes kiveetel, 1. kurs... 357 o., 3. p.

Majd új sorra tért át, és képezte a következő alakot.

Minden további nélkül úgy lehetett volna utasítani, hogy ne rögtön gépeljen, hanem egy az adatszalaghoz hasonló eredményszalagra lyukasztva hozza az eredményt (annak az az előnye, hogy az eredményeket újra és újra le lehet gépeltetni egy telexen) — konkrétan a mi gépünk esetében ilyenkor a fentebb említett 626 helyett mindenütt ezt kellett volna írnom: 526. Már lényegesebb programmodosítás kellene ahhoz, hogy a gép ne egyenként olvassa be az alakokat, hanem beolvassa az egész adatszalogot, majd ennek végeztével kezdjen hozzá a kiíráshoz — és így tovább.

6. De mire jó mindez? — kérdezhetné valamely szkeptikus olvasóm. Van az ilyen munkáknak elvi haszna (jobban megismerjük a gépet, az emberi gondolkodást, a nyelvtant), gyakorlati is (ha a gép — például — oroszra fordít majd, akkor természetesen ismernie kell az itt tárgyalt apró grammatikai kérdést is; a gépi tapasztalatok visszaalkalmazhatók az emberek oktatására stb.).

De, bevallom, engem e munkám során főleg nem e hasznok vezéreltek. Engedje meg az olvasó, hogy általam itt feltételezett kérdésre visszakérdezzek: És ön miért olvasta el e kis cikket? Volt valami konkrét haszna belőle? Persze, talán volt; vagy van, akinek volt. Ám feltehetően a többség konyhájára semmiféle hasznót sem hoztam e soraimmal — csupán érdekesekek voltak azok az olvasó számára, ha már idáig jutott bennük. Nos, végső válaszom nekem is ez: egyszerűen érdekes volt számomra együtt dolgozni a géppel. Mivel gondoltam, mást is érdekelhet — el is mondtam belőle egy-két apróságot a fentiekben.

Papp Ferenc