

Kompiuterinių matematikos sistemų programavimo kalbų lyginamoji analizė

Sigita TURSKIENĖ (ŠU)

el. paštas: sigita@fm.su.lt

Įvadas

Kompiuterinės matematikos sistemos (KMS) atsirado 7 dešimtmetyje, kaip alternatyva tik skaičiuojamąsias galimybes turinčioms universalioms programavimo kalboms (UPK). Sparčiai plito KMS versijos, atsirado daug interpretatorių įvairių tipų kompiuteriams. Lietuvoje išivyravo sistemos: MAPLE, MATHCAD, MATLAB, MATHEMATICA.

[1, 4, 6, 7] darbuose KMS vartojamos kaip analizinių skaičiavimų sistemos. Išsamiau KMS programavimo kalbos apžvelgiamos [3, 5]. UPK leksika nagrinėjama [2]. Tyrimus būtina tęsti. Pirma priežastis – pastovus KMS tobulėjimas ir naujų atsiradimas. Antra priežastis – Lietuvoje nėra išsamių tyrimų, kurie įvertintų KMS programavimo kalbų galimybes lyginant su UPK. Pavienių tyrimų yra mažai ir jų rezultatus sunku apibendrinti.

Į nagrinėjamų kalbų sąrašą įtraukėme programavimo kalbas: C/C++, PASCAL, MATHEMATICA 5, MATHCAD 2001, MAPLE 8. PASCAL įtraukėme dėl to, kad programavimui mokytis skirta kalba ir plačiai naudojama mokykloje. C/C++ daugiau skirta profesionalams ir plačiai naudojama Lietuvos aukštosiose mokyklose. Be to, šios kalbos sudaro daugiausiai vartojamų procedūrinio ir objektinio programavimo kalbų grupę. Pasirinktas kalbų lyginimo matas yra kiek primityvus, nes jų paskirtys skirtingos. PASCAL kalba rodo taisyklingas programavimo konstrukcijų vartojimo galimybes, fundamentalioms programavimo sąvokoms suvokti. KMS kalbos yra į problemą orientuotos programavimo kalbos, t.y., skirtos matematikos uždaviniams spręsti. Jų naudojimo sritys yra siauresnės, nemažas dėmesys skiriamas rezultatų vizualumui.

Darbo tikslas – atlikti KMS programavimo kalbų galimybių analizę, atskleidžiant kalbų sintaksę, panašumus ir skirtumus. Išsiaiškinti, kiek jų žymenis atitinka UPK vartojamus žymenis. Vartotojai tokius žymenis lengviau išimena, suderina su žinomomis programavimo kalbomis. Tai palengvina KMS taikymą darbui ir mokymui(si).

Populiarių KMS programavimo kalbų pagrindinės charakteristikos

Visos KMS turi vidines aukšto lygio programavimo kalbas. KMS surinko kelių UPK (BASIC, PASCAL, FORTRAN, C) savybes. Todėl KMS kalbomis galima realizuoti

kelias programavimo paradigmas: funkcinę, struktūrinę-modulinę, objektinę, loginę, procedūrinę. Paskutinės KMS realizacijos leidžia organizuoti vaizdinį programavimą.

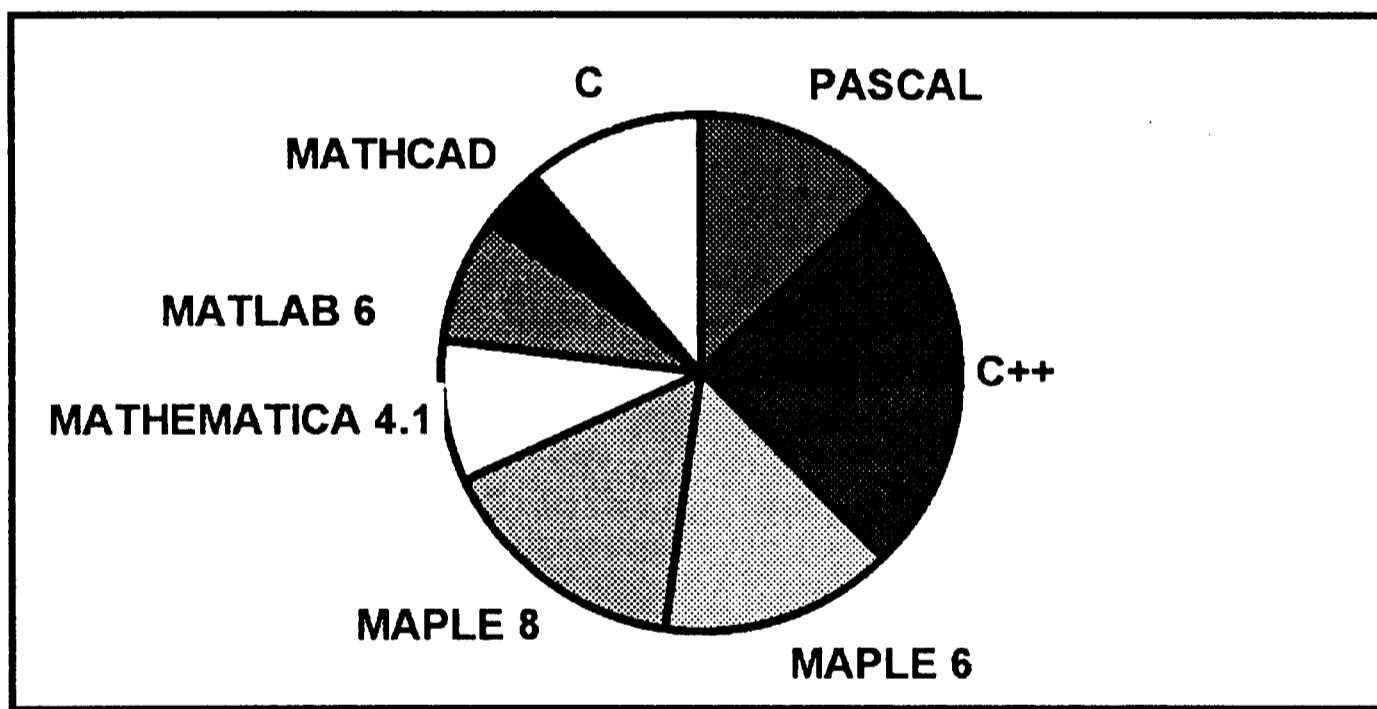
Programų skaitiniams duomenims apdoroti kūrimas KMS kalbomis mažai skiriasi nuo analogiškų veiksmų UPK. Veiksmų su analiziniais duomenimis programavimas skiriasi nuo įprasto programavimo, nes čia reikia daugiau tikrinti reiškinių tipus ir struktūrą.

Palyginę PASCAL, C/C++, MAPLE, MATHEMATICA, MATLAB kalbų abėcėlių simbolius [2, 5], pastebime, kad C++ kalba yra turtingesnė. KMS kalbomis sunkiau realizuoti tradicinio programavimo uždavinius, nes jose yra mažiau specialių ženklų ir bazinių žodžių (1 pav.). Kalbų bazinių žodžių sutapimas parodytas 1 lentelėje.

KMS vardų sudarymo taisyklės panašios į PASCAL ir C kalbų vardų sudarymo taisyklės [2, 5]. MAPLE ir MATHEMATICA varduose, kaip C kalboje, didžiosios ir mažosios raidės yra skirtingos. MATHEMATICA vartotojo vardas būtinai prasideda mažąja raide. Tai apsunkina pradinį programavimo mokymąsi.

Palyginus konstantų tipus [2, 5] pastebime, kad kalbos turi tuos pačius pagrindinius konstantų tipus: sveikąjį, realųjį, simbolinį, loginį, simbolių eilutę. Konstantų užrašymo sintaksė ir žodžiai yra panašūs.

Kintamųjų aprašai šiek tiek skiriasi nagrinėjamose kalbose, nors prasmė išlieka ta pati. Beveik visose kalbose turi būti nurodytas tipas ir kintamojo vardas. Iš anksto



1 pav. Programavimo kalbų baziniai žodžiai.

1 lentelė. Programavimo kalbų bazinių žodžių aibių sankirtos (procentais).

Programavimo kalba	MAPLE	MATLAB	MATHEMATICA
PASCAL	37,1	34,3	28,6
C++	10,8	12,2	16,2
MAPLE	100	30,4	30,4
MATHEMATICA	56	56	100
MATLAB	58,3	100	58,3

apibrėžti kintamojo tipo nereikia MATLAB kalba. Tipas apibrėžiamas kintamajam priskiriant reikšmę. Tai supaprastina programų kūrimą.

Pirmosiose KMS versijose visas dėmesys skiriamas valdymo konstrukcijoms, mažai kalbama apie svarbų procedūrinių kalbų komponentą – duomenų struktūras. Rašant programas KMS kalbomis, būtina duomenų struktūras taikyti ne tik elementariu lygiu, bet išnaudoti jų abstrakcijos lygį. Jos sudaromos arba reiškinių specialiomis sintaksinėmis konstrukcijomis (seka, sąrašas, aibė) arba kreipiniu į specialią funkciją – konstruktorių (masyvas, lentelė). MAPLE kalba turi daugiau bazinių duomenų tipų [5] negu kitos programavimo kalbos. C kalba turi daugiau sveikojo tipo atmainų, turi papildomą atmainų grupę sveikiems skaičiams be ženklo, ko nėra PASCAL kalba. Masyvų aprašų sintaksė yra skirtinga visomis KMS kalbomis. C kalba masyvų aprašas yra trumpiausias.

Nors abstraktieji duomenų tipai yra objektinio programavimo pagrindas, tačiau beveik visos KMS kalbų realizacijos turi duomenų objektus ir tai laikoma natūraliu dalyku. Pvz., MATLAB 6 kalba [5] yra 7 pagrindiniai klasių objektai.

Aritmetinių operacijų žymenys yra panašūs visose kalbose (2 lentelė). Operacijų atlikimo tvarka nevienoda KMS kalbomis. Pvz., MATLAB kalba [5] loginių operacijų rangas aukštesnis negu aritmetinių. Skirtingai nuo kitų programavimo kalbų, MATLAB kalbos aritmetinių ir palyginimo operacijų taikymo spektras yra daug platesnis, jos skirtos darbui ne tik su paprastais dydžiais, bet ir vektoriais, matricomis, simbolių eilutėmis. KMS kalbų aritmetinės ir palyginimo operacijos [5] turi analogiškas funkcijas. Tai užtikrina operatorinę ir funkcinę programavimą. C kalba turi daugiau vienviečių operacijų. MATHEMATICA kalbos sutrumpinta aritmetinių operacijų (++, –) forma leidžia trumpinti matematinių reiškinių užrašus, nors aiškumas kažkiek mažėja. MATLAB 6 kalba turi mažiau loginių operacijų negu kitos kalbos. Loginių operacijų operandais gali būti skaičiai ir simbolių eilutės. Loginės operacijos [5] užrašomos skirtingai įvairiose programavimo kalbose. MAPLE kalba vartojama trireikšmė logika: true, false, FAIL. Skliaustų gylis reiškiniuose praktiškai neribojamas.

Programuotojas savo veiksmus kompiuteriui užrašo sakiniiais. Priskyrimo sakinyje užrašomas panašiai visose kalbose, skiriasi priskyrimo operacijos ženklas (= :=).

2 lentelė. Aritmetinių operacijų žymenys programavimo kalbose

Operatorius	MATLAB	MATHEMATICA	MAPLE	PASCAL	C
Sudėtis	+	+	+	+	+
Atimtis	–	–	–	–	–
Daugyba	*	* arba tarpas	*	*	*
Elementinė daugyba	.*				
Kėlimas laipsniu	^	^	** arba ^		
Elementinis kėlimas laipsniu	.^				
Dešininė dalyba	\				
Dalyba	/	/	/	/	/
Elementinė dalyba	./				
Dalybos liekana			Mod	Mod	%
Sveikoji dalyba				Div	

KMS populiariausias sąlygos sakinytis yra if sakinytis. Sąlygos sakinių [5] užrašymo sintaksė panaši visose kalbose. C kalba nėra žodelio then. MATHEMATICA kalba yra keli sąlygos sakiniai ir jų sintaksė skiriasi nuo tradicinių sąlygos sakinių if. Ne visos KMS turi daugiavariantinių alternatyvų sakinių. MAPLE 8 kalba neturi tokio sakinio. MATLAB ir MATHEMATICA kalbos turi switch sakinių. Nagrinėjamos kalbose variantinio sakinio veikimo principas yra toks pat, bet sakinių baziniai žodžiai ir sintaksė skiriasi.

C, PASCAL kalbos turi tris ciklo sakinius. Sakinio while sintaksė panaši PASCAL, C, MAPLE kalbomis. MATLAB kalbos sintaksė leidžia ciklo parametru naudoti masyvą. MATLAB kalba neturi do sakinio. MAPLE kalbos sąlygos ir ciklo sakiniai turi specifinius pabaigos žodelius: end if, end do.

KMS programavimo kalbos leidžia rašyti dideles programas matematiniamis uždaviniais spręsti. Programuotojas susiduria su programų teisingumo nustatymo problema, nes jose didėja klaidų galimybė. KMS klaidų paieškos ir taisymo procesą palengvina geros programų derinimo priemonės [5]. Palyginus programų derinimo priemones, pastebimi MAPLE pranašumai: didesnės ir įvairesnės galimybės, rezultatai pateikiami suprantamesne forma, kelios priemonės gana originalios ir jų neturi net UPK. MAPLE 8 turi klaidų gaudyklę try-catch (panaši konstrukcija yra C++, JAVA).

Nagrinėjamų KMS kalbų bazinių žodžių, pagrindinių konstrukcijų, operacijų žymenų analizė [2, 5] rodo, kad pirmųjų sistemos MAPLE versijų programavimo kalba buvo artimesnė PASCAL kalbai, paskutiniųjų – artimesnė C kalbai. MATHEMATICA kalba artima C kalbai. Ankstesnės sistemos MATHCAD versijos neturėjo įprasto programavimo priemonių. Buvo galima kurti tik tiesines programas, realizuojant funkcinių programavimą. Programiniai moduliai kuriami nuo MATHCAD PLUS 6.0 versijos. MATHCAD 2000/2001 versijoje įvestos būtinos programavimo priemonės.

KMS kalbų ir UPK palyginimas

Išsamus KMS programavimo galimybių nagrinėjimas užima daug laiko. Be to, kai kurios sistemų savybės išryškėja tik detalai su jomis susipažinus. Kad palengvinti KMS programavimo galimybių išsivadinimą, pateiksime nagrinėjamų KMS (MAPLE, MATLAB, MATHEMATICA) ir UPK (C, PASCAL) dar neaptartus panašumus ir skirtumus. KMS kalbų ir UPK skirtumai:

1. KMS dažniausiai atliekamas komandų interpretavimas, o UPK parašytos programos dažniausiai yra kompiliuojamos. Sukompiliuotos programos vykdomos greičiau, bet kurti ir derinti lengviau interpretuojamas programas. Sistemose MATHEMATICA ir MATLAB yra galimybė sukompiliuoti atskirus reiškinius.
2. UPK neturi gerų priemonių simboliniams uždaviniais spręsti. Tokias galimybes siūlo KMS: reiškinių pertvarkiai, diferencijavimas, integravimas ir pan.
3. KMS kalbomis lengviau aprašomi ir greičiau skaičiuojami matematiniai reiškiniai. C, PASCAL kalbomis užrašyti reiškiniai nėra informatyvūs, sunkiau skaitomi.
4. KMS turi daugiau duomenų tipų. Pvz., kompleksinių skaičių tipą. Sistemos MATLAB duomenų tipų hierarchijos viršuje yra masyvo tipas array.
5. KMS turi žymiai daugiau standartinių funkcijų (kelis tūkstančius) negu UPK. Todėl nereikia daug algoritmų programuoti. Pvz., lengvesnis darbas su masyvais.

6. KMS kalbomis sukurtas programos galima versti į C ir FORTRAN kalbų kodus. Tai įgalina sudėtingus matematinius reiškinius užrašyti KMS kalba, juos pertvarkyti, konvertuoti į C kodą ir perkelti į rašomą programą. Tačiau ne visos KMS kalbų instrukcijos kompiliuojamos. MATLAB kalba kodo transformavimas yra paprastesnis nei MAPLE kalba. Sistema MAPLE neturi galimybės lengvai kurti vykdomų exe failų.
7. KMS saugo daug matematikos, fizikos, chemijos konstantų. Yra begalybės konstanta. Tai palengvina fizikinių reiškinių kompiuterinių modelių kūrimą.

KMS ir UPK panašumai:

1. KMS kalba, kaip ir kitos kalbos, sudaroma iš duomenų tipų, operatorių, funkcijų, procedūrų ir valdymo struktūrų.
2. KMS kalbos turi tuos pačius pagrindinius duomenų tipus, kaip UPK. Simbolių eilutė traktuojama kaip vienmatis simbolių masyvas.
3. Įgalina kurti šablonus atstovaujančius visai reiškinių klasei.
4. Įgalina realizuoti rekursyviuosius ir rekurenčiuosius algoritmus.
5. Turi programos vykdymo reguliavimo komandas (abort, break, continue, return).
6. Turi nukreipimo sakinių go to. Išimtis MATLAB kalba – orientuota į struktūrinį programavimą.
7. Įgalina kurti funkcijas ir procedūras, kaip savarankiškus programinius modulius. Turi priemones vartotojo paketams ir bibliotekoms kurti.
8. Turi priemones komentarams rašyti programoje.
9. Turi darbui su failais priemones. Sistemose MATLAB ir MAPLE realizuotos duomenų skaitymo/rašymo instrukcijos panašios į C kalbos instrukcijas.

Išvados

1. Nagrinėtos KMS turi vidines aukšto lygio programavimo kalbas, kurių pagrindinės konstrukcijos yra artimos UPK.
2. Išmokti programuoti KMS kalbomis nesunku, jeigu yra pradinės programavimo žinios bet kuria universaliąja programavimo kalba.
3. C/C++ programavimo kalba gerai dera su KMS kalbomis. Kaip rodo darbo praktika, tokia programinių priemonių simbiozė duoda gerus darbo rezultatus.
4. Nėra pakankamai mokymo priemonių lietuvių kalba.

Literatūra

1. A. Domarkas, R. Rakauskas ir kt. Kompiuterinės algebros ir skaitinių metodų sąsaja, *Liet. matem. rink.*, **41**(spec. nr.), 184–190 (2001).
2. G. Grigas, Programavimo kalbų leksikos elementų analizė mokymo požiūriu, *Informatika*, **35**, 45–67 (2000).
3. J. Lipeikienė, *Matematika su kompiuteriu*, Vilnius (2002).
4. S. Turskienė, Kompiuterinės matematikos sistemos mokymo procese, *Liet. matem. rink.*, **42**(spec. nr.), 283–288 (2002).
5. S. Turskienė, *Kompiuterinių matematikos sistemų programavimo galimybės*, Šiauliai (2004).
6. S. Turskienė, Computer technology and teaching mathematics in secondary schools, *Informatics in Education*, **1**, 149–156 (2002).
7. S. Turskienė Kompiuterinės matematikos sistemų taikymas fizikos mokymui, *Informacijos mokslai*, **26**, 73–76 (2003).

SUMMARY

S. Turskienė. Comparative analysis of programming languages possibilities of computer mathematics systems

The paper analyses possibilities of programming languages of computer mathematics systems. The summary table of operations of 5 programming languages is presented. The advantages and disadvantages between programming languages of computer mathematics systems (MAPLE, MATHEMATICA, MATLAB) and professional programming languages (C/C++, PASCAL) are presented, too. That makes easier the application of computer mathematics systems in practice.

Keywords: programming language, computer mathematics system.