

VILNIAUS UNIVERSITETAS  
MATEMATIKOS IR INFORMATIKOS FAKULTETAS  
INFORMATIKOS KATEDRA

Baigiamasis bakalauro darbas

## **Klaviatūros tvarkyklių tyrimas**

(Analysis of Keyboard Layouts and Drivers)

Atliko: 4 kurso 8 grupės studentas

Rimas Kudelis

Darbo vadovas:

doc. dr. Vladas Tumasonis

Vilnius

2009

## Turinys

Ivadas.....	3
1. Pagrindinės tvarkyklių savybės.....	4
1.1. Kirčiuotųjų raidžių įvedimas.....	4
1.1.1. Tęsties ženklai.....	4
1.1.2. Kombinacinių diakritinių ženklų įvedimas.....	6
2. „Windows“ tvarkyklės.....	6
2.1. Kirčiuotųjų raidžių įvedimas.....	8
2.1.1. Tęsties ženklai.....	8
2.1.2. Kombinacinių diakritinių ženklų įvedimas.....	8
2.2. Klaviatūros išdėstymo aprašo kūrimas.....	10
3. „X Window System“ tvarkyklės.....	10
3.1. Kirčiuotųjų raidžių įvedimas.....	11
3.1.1. Tęsties ženklai.....	12
3.1.2. „Compose“ klavišas.....	12
3.1.3. Kombinacinių diakritinių ženklų įvedimas.....	15
3.2. Klaviatūros išdėstymo aprašo kūrimas.....	15
3.3. Komponavimo lentelė ir jos kūrimas.....	18
4. „Mac OS X“ tvarkyklės.....	19
4.1. Kirčiuotųjų raidžių įvedimas.....	21
4.1.1. Tęsties ženklai.....	21
4.1.2. Kombinacinių diakritinių ženklų įvedimas.....	22
4.2. Klaviatūros išdėstymo aprašo struktūra ir kūrimas.....	22
Išvados ir pasiūlymai.....	30
Analysis of Keyboard Layouts and Drivers (Summary in English).....	31
Literatūros sąrašas.....	32
1, 2, 3, 4, 5, 6, 7, 8 priedai	

## Ivadas

Klaviatūra nuo pat jos sukūrimo yra pagrindinis įrankis tekstinei informacijai į kompiuterį įvesti. Tačiau pati klaviatūra šios (tekstinės) informacijos į kompiuterį nesiunčia – ji tik perduoda signalus apie klavišų nuspaudimus. Šiuos signalus kompiuteryje apdoroja speciali programa – klaviatūros tvarkyklė. Šiame darbe siekiama palyginti klaviatūrų tvarkyklių galimybes populiariose operacinėse sistemose – *Windows*, *Mac OS X* ir *Linux* (pastarajai OS atstovaus *X Window System* grafinė aplinka, dar vadinama *X11*).

Kadangi akivaizdu, kad pagrindinės tvarkyklių savybės yra vienodos, šiame darbe joms nebus skiriama daug dėmesio. Daugiausia jo bus kreipiamas į rečiau naudojamas galimybes – kirčiuotųjų raidžių įvedimą bei apskritai galimybę išplėsti klaviatūra įvedamų simbolių aibę.

Poreikis tokiam palyginimui kyla iš fakto, jog bendrinė lietuvių kalba turi net 68 kirčiuotas raides. Šios raidės gali būti pavartotos žodynuose, enciklopedijose, vadovėliuose, tekstynuose, terminologinėse duomenų bazėse ir pan., o kartais jų prisireikia net ir „paprastame“ tekste, norint išvengti dviprasmybių. 33 jų turi apibrėžtas pozicijas Unikodo lentelėje (Unikodo kodus), o kitas galima išreikšti vadinamosiomis kompozicinėmis sekomis. Tokią seką sudaro pamatinė raidė (nagrinėjamu atveju – nekirčiuotoji raidė) ir kombinacinis diakritinis ženklas (arba tiesiog kirtis). Šioje taisyklėje yra ir ypatingų atvejų – tai mažųjų raidžių „i“ ir „j“ bei jų vedinių kirčiavimas. Unikodo standarte apibrėžta, jog šias raides kombinuojant su virš jų atsiduriančiais diakritiniais ženklais, jos turi netekti savo taškų. Dėl to, kombinuojant lietuviškas kirčiuotas raides „i“, „j“ ir „j“, šis taškas jau kaip kombinacinis diakritinis ženklas joms turi būti uždedamas papildomai. Kai tekste norima pabrėžti kombinacinių ženklų nesavarankiškumą, jie dedami ant „šabloninio“ simbolio ◌ (pvz. ◌̇).

Pastebėtina, jog kirčiuotas raides, neturinčias apibrėžtų Unikodo pozicijų, galima išreikšti ne tik kompozicinėmis sekomis, bet ir Unikodo Privačiosios erdvės kodais. Privačioji erdvė – tai režis Unikodo kodų, paliktų privačioms reikmėms ir neturinčių standartizuotos semantinės prasmės. Vis dėlto, kadangi šioje erdvėje kiekvienas šriftų kūrėjas gali „tvarkytis“ savo nuožiūra, joje neįmanoma tikėtis suderinamumo, tad naudoti Privačiosios erdvės kodus, ypač elektroninėje terpėje, būtų netoliaregiška. Kita vertus, Privačiojoje erdvėje iš anksto sukomponuotos kirčiuotosios raidės gali labai praversti, rengiant maketus spaudiniams.

Privačiosios erdvės kodų įvedimas šiame darbe nebus nagrinėjamas atskirai, nes tai tėra atskiras nagrinėjamų principų panaudojimo atvejis.

## 1. Pagrindinės tvarkyklių savybės

Kaip ir galima tikėtis, pagrindinės visų tvarkyklių savybės yra vienodos, o pateikiami išdėstymai labai panašūs. Pasitaikantys nedideli kiekvienos aplinkos ypatumai bus paminėti tolesniuose skyriuose.

### 1.1. Kirčiuotųjų raidžių įvedimas

Visose tirtose aplinkose kirčiuotąsias lietuviškas raides galima įvesti mažiausiai dviem būdais:

- per tęsties ženklus,
- tiesiogiai įvedant kombinacinius diakritinius ženklus.

Žemiau jie abu trumpai aptarti.

#### 1.1.1. Tęsties ženklai

Tęsties ženklas (angl. *dead key*) – tai toks klaviatūros ženklas, kurį surinkus nieko negaunama: jo veikimas pasireiškia surinkus kitą ženklą. Tai jau senokai žinomas tiesioginis klaviatūros galios išplėtimas, nereikalaujantis nei papildomų fizinių klavišų klaviatūroje, nei papildomo šios graviravimo, nei komplikuoto klavišų sekų įsiminimo [Tum07]. Klaviatūroje tęsties ženklas paprastai būna patalpinamas vietoje atitinkamo ar panašaus įprasto ženklo. Naudojant tęsties ženklą, kirčiuotoji raidė gaunama paeiliui klaviatūra spaudžiant atitinkamus kirtį bei raidę, ant kurios šis turėtų būti uždedamas. Pavyzdžiui, simbolis „à“ (mažoji raidė a su kairiniu kirčiu) gali būti gaunamas paeiliui spaudžiant tęsties ženklo – kairinio kirčio – klavišą (‘) bei pamatinę raidę „a“. Visų lietuviškų kirčiuotų raidžių įvedimo per tęsties klavišus etaloninis variantas formalizuotai aprašytas 1 lentelėje.

Norint įvesti kirčiuotas lietuvių kalbos raides, tęsties ženklais turėtų tapti trys ženklai – <^>, <’> ir <~>. Jei norima įvesti ir kitų kalbų raides su diakritiniais ženklais, tęsties ženklų reikėtų daugiau. Pavyzdžiui, norint rašyti prancūziškai ir vokiškai, tęsties ženklais galėtų būti paversti ir ženklai <^> (atitiktų stogelį) ir <~> (atitiktų umliautą). Tiesa, prancūzų kalboje naudojama dar ir raidė „ç“, tačiau kadangi tai yra vienintelė šiuo atveju reikalinga raidė su sedile, ją galima priskirti kuriai nors kitai nenaudojamai sekai, pavyzdžiui <^>, <c>.

Reikia pastebėti, jog kiekvienas tęsties ženklas „kainuoja“, nes, įtraukiamas į klaviatūrą,

Paeiliui spaudžiami klavišai	Gau- nama raidė	Gaunamos raidės		Paeiliui spaudžiami klavišai	Gau- nama raidė	Gaunamos raidės	
		unikodinė išraiška	dekom- pozicija			unikodinė išraiška	dekom- pozicija
<^>, <A>	À	00C0	À	<^>, <Y>	Ý	00DD	Ý
<^>, <a>	à	00E0	à	<^>, <y>	ý	00FD	ý
<^>, <A>	Á	00C1	Á	<~>, <Y>	Ÿ	1EF8	Ÿ
<^>, <a>	á	00E1	á	<~>, <y>	ÿ	1EF9	ÿ
<~>, <A>	Ã	00C3	Ã	<~>, <J>	Ĵ	004A 0303	Ĵ
<~>, <a>	ã	00E3	ã	<~>, <j>	ĵ	006A 0307 0303	ĵ
<^>, <Ą>	Ą	0104 0301	Ą	<~>, <L>	Ł	004C 0303	Ł
<^>, <a>	ą	0105 0301	ą	<~>, <l>	ł	006C 0303	ł
<~>, <Ą>	Ą	0104 0303	Ą	<~>, <M>	Ŭ	004D 0303	Ŭ
<~>, <a>	ą	0105 0303	ą	<~>, <m>	ŭ	006D 0303	ŭ
<^>, <E>	È	00C8	È	<~>, <N>	Ñ	00D1	Ñ
<^>, <e>	è	00E8	è	<~>, <n>	ñ	00F1	ñ
<^>, <E>	É	00C9	É	<^>, <O>	Ò	00D2	Ò
<^>, <e>	é	00E9	é	<^>, <o>	ò	00F2	ò
<~>, <E>	Ê	1EBC	Ê	<^>, <O>	Ó	00D3	Ó
<~>, <e>	ê	1EBD	ê	<^>, <o>	ó	00F3	ó
<^>, <Ę>	Ę	0118 0301	Ę	<~>, <O>	Õ	00D5	Õ
<^>, <ę>	ę	0119 0301	ę	<~>, <o>	õ	00F5	õ
<~>, <Ę>	Ę	0118 0303	Ę	<~>, <R>	Ř	0052 0303	Ř
<~>, <ę>	ę	0119 0303	ę	<~>, <r>	ř	0072 0303	ř
<^>, <Ė>	Ė	0116 0301	Ė	<^>, <U>	Ù	00D9	Ù
<^>, <ė>	ė	0117 0301	ė	<^>, <u>	ù	00F9	ù
<~>, <Ė>	Ė	0116 0303	Ė	<^>, <U>	Ú	00DA	Ú
<~>, <ė>	ė	0117 0303	ė	<^>, <u>	ú	00FA	ú
<^>, <I>	Ì	00CC	Ì	<~>, <U>	Ũ	0168	Ũ
<^>, <i>	ì	0069 0307 0300	ì	<~>, <u>	ũ	0169	ũ
<^>, <I>	Í	00CD	Í	<^>, <Ų>	Ų	0172 0301	Ų
<^>, <i>	í	0069 0307 0301	í	<^>, <ų>	ų	0173 0301	ų
<~>, <I>	Ĩ	0128	Ĩ	<~>, <Ų>	Ų	0172 0303	Ų
<~>, <i>	ĩ	0069 0307 0303	ĩ	<~>, <ų>	ű	0173 0303	ű
<^>, <Į>	Į	012E 0301	Į	<^>, <Ū>	Ū	016A 0301	Ū
<^>, <į>	į	012F 0307 0301	į	<^>, <ū>	ú	016B 0301	ū
<~>, <Į>	Į	012E 0303	Į	<~>, <Ū>	Ū	016A 0303	Ū
<~>, <į>	į	012F 0307 0303	į	<~>, <ū>	ű	016B 0303	ű

1 lentelė. Kirčiuotų lietuviškų raidžių įvedimas, naudojant tęsties ženklus

jis paprastai blokuoja joje esantį savarankišką ženklą, kuris tuomet paprastai gaunamas jau dviem klavišų paspaudimais. Pavyzdžiui:

- <`>, <tarpas> → `
- <~>, <tarpas> → ~
- <'>, <tarpas> → '

### 1.1.2. Kombinacinių diakritinių ženklų įvedimas

Antrasis būdas – tai tiesioginis kombinacinių diakritinių ženklų įvedimas klaviatūra. Tam reikia norimiems ženklams tiesiog suteikti vietas klaviatūroje (ar klavišų kombinacijose su tęsties ženklais). Šis būdas turi savų privalumų ir trūkumų.

Didžiausias jo privalumas – tai galimybė bet atskirų aprašų neribotai komponuoti simbolius su bet koku kirčių skaičiumi ar tvarka. T.y., šiuo būdu įmanoma įvesti praktiškai bet ką. Todėl ši galimybė yra labai pravarti lingvistams. Naudojant šį metodą, galimų įvesti simbolių aibės „ribotuvu“ galima laikyti nebent pačių raidžių kiekį – pavyzdžiui, lietuviškoje klaviatūroje paprastai nėra pasiekiami rusiški ar graikiški rašmenys.

Šio metodo trūkumas – tai jo santykinis sudėtingumas ir neatsparumas klaidoms. Kadangi daugumoje aplinkų kompozicinės raidės iki šiol atvaizduojamos prastai, vartotojui reikėtų pačiam sekti savo klaidas, pavyzdžiui, nepamiršti sudėti kombinacinių taškų ant lietuviškų raidžių „i“, „j“ ir „j“, prieš dedant ant jų kirtį (palyginimui: didžioji raidė J su riestiniu kirčiu būtų komponuojama iš dviejų dalių: J̣; tuo tarpu mažoji – iš trijų: j̣̣̣). Kitas trūkumas – tai faktas, kad, renkant raidę šiuo būdu, ji *visada bus komponuojama* iš raidės ir kombinacinio diakritinio ženklo – net ir tada, kai jau sukomponuotas jos variantas turi savo vietą unikode. Trečiasis trūkumas – tai, jog šis būdas reikalauja kiekvienam norimam įvesti diakritiniam ženklui suteikti atskirą vietą klaviatūroje. Pavyzdžiui, net jei rašančiajam tereikia vieno simbolio su sedile (tarkime, minėtosios raidės „ç“), norint ją išgauti jau tekstų skirti atskirą klaviatūros klavišą šiam diakritiniam ženklui įvesti.

## 2. „Windows“ tvarkyklės

*Windows* – vienintelė iš tirtų sistemų, kurioje klaviatūros išdėstymų tvarkyklės turi būti kompiliuojamos, todėl jas kuriant, neįmanoma apsieiti be specialių programų. Tokią programą – *Microsoft Keyboard Layout Creator* (toliau – *MSKLC*) galima parsisiųsti

nemokamai iš *Microsoft* svetainės [Mic09].

Pažymėtina, jog *Windows* atveju sistemos vartotojas negali ne tik įdiegti išdėstymo, neturėdamas kompiuterio administratoriaus teisių, bet ir konfigūruoti jau įdiegto išdėstymo niuansų; jam leidžiama tik keisti naudojamų išdėstymų sąrašą. Taigi, kuriame išdėstymo konfigūraciją atlikti privalo išdėstymo autorius. O šioji gan skurdi – leidžiama keisti tik du parametrus:

1. ar *Lyg2* (*Shift*) klavišo paspaudimas turi nutraukti didžiųjų raidžių rašymą;
2. ar *Lyg2* ir *Naikinti* (*Backspace*) klavišų kombinacijos turi įterpti teksto krypties žymeklius.

Įdomus pastebėjimas – *Windows* klaviatūros išdėstymo autorius pats gali pasirūpinti jo pavadinimo vertimu. Nors pačioje *MSKLC* programoje tokios galimybės šiuo metu nėra, tereikia ja sukurtą failą atverti grynojo teksto redaktoriumi ir jame įrašyti norimus vertimus. Tuomet, kitąkart kompiliuojant išdėstymą, jis bus sukompiliuotas jau su vertimais [Kap07].

Kiekvienam grafinio ženklo klavišui (o taip pat tarpo klavišui) leidžiama priskirti iki aštuonių skirtinguose lygiuose įvedamų simbolių, gaunamų spaudžiant šį klavišą skirtingomis aplinkybėmis (vadinsime jas lygiais, pagal numerį sąrašė):

1. be modifikatorių;
2. laikant nuspaustą *Lyg2* klavišą;
3. laikant nuspaustą *Lyg3* (*AltGr*) klavišą (beje, jo veikimą *Windows* papildomai emuliuoja, laikant vienu metu nuspaustus *Vald* (*Ctrl*) ir *Alt* klavišus);
4. laikant nuspaustus *Lyg2* bei *Lyg3* klavišus;
5. laikant nuspaustą *Vald* klavišą;
6. laikant nuspaustus *Lyg2* bei *Vald* klavišus;
7. aktyvavus *Didž* klavišą;
8. aktyvavus *Didž* klavišą bei laikant nuspaustą *Lyg2* klavišą.

Pirmuose šešiuose lygiuose vienu klavišo paspaudimu leidžiama įvesti iki keturių unikodo kodų, tad juose galima naudoti ir kompozicines sekas. Likusiuose dviejuose šios prabangos nėra – čia vienu klavišo paspaudimu galima įvesti tik vieną unikodo kodą.

Nors tai gali atrodyti nemažai, tačiau net *MSKLC* žinynas nepataria naudoti kitų, nei pirmojo–ketvirtojo lygių. Penktasis ir šeštasis lygiai nenaudotini todėl, kad klavišų kombinacijas su *Vald* klavišu absoliuti dauguma programų apdoroja kaip valdymo komandas, tad įvestas tekstas tiesiog nepasiektų lauko, į kurį jis įvedamas. Septintasis ir aštuntasis lygiai nepatartini tiesiog dėl to, kad jų naudojimas veikiausiai griautų nusistovėjusius vartotojo įpročius ir lūkesčius. Šie du lygiai šiuo metu naudojami tik šveicarų ir hebrajų išdėstymuose.

## 2.1. Kirčiuotųjų raidžių įvedimas

Kirčiuotąsias raides *Windows* aplinkoje galima įvesti dviem būdais:

- per tęsties ženklus,
- tiesiogiai įvedant kombinacinius diakritinius ženklus.

### 2.1.1. Tęsties ženklai

Tęsties ženklą klavišui galima priskirti bet kuriame iš išvardintų lygių. Deja, ši galimybė ribota: tiek tęsties ženklas, tiek simbolis, su kuriuo jis kombinuojamas, tiek galutinis gaunamas simbolis turi užimti tik vieną unikodo poziciją. Dėl to *Windows* atveju tęsties ženklais galimų įvesti lietuviškų kirčiuotųjų raidžių aibė (atmetus Privačiosios erdvės kodus) susiaurėja iki 33 raidžių, turinčių atskirus unikodo kodus. Ji pateikta 2 lentelėje. Likusias kirčiuotas raides tenka įvesti antruoju būdu.

*Windows* tvarkyklėse įmanoma aprašyti ir keleto paeiliui įvedamų tęsties ženklų veikimą. Deja, tai ne tik nerealizuota *MSKLC* programoje, bet ir (dėl ką tik paminėto apribojimo) nepadėtų įvesti unikodo pozicijų neturinčių raidžių su dviem ar daugiau diakritiniais ženklais [Kap05].

Taigi, bendruoju atveju įvedimą naudojant tęsties klavišus *Windows* sistemoje galima aprašyti tokiu formalizmu:

$$d_1 c_1 \rightarrow k_1 \quad ,$$

čia  $d_1$  – tęsties ženklas,  $c_1$  – įprastas ženklas,  $k_1$  – atskirą unikodo kodą turintis simbolis.

### 2.1.2. Kombinacinių diakritinių ženklų įvedimas

Šiuo būdu – įvedant atskirai raidę ir reikiamus kombinacinius diakritikus (jų įvedimo



Paeiliui spaudžiami klavišai	Gaunama matoma raidė	Gautos raidės Unikodinė išraiška	Paeiliui spaudžiami klavišai	Gaunama matoma raidė	Gautos raidės Unikodinė išraiška
<^>, <A>	À	00C0	<~>, <N>	Ñ	00D1
<^>, <a>	à	00E0	<~>, <n>	ñ	00F1
<^>, <A>	Á	00C1	<^>, <O>	Ò	00D2
<^>, <a>	á	00E1	<^>, <o>	ò	00F2
<~>, <A>	Ã	00C3	<^>, <O>	Ó	00D3
<~>, <a>	ã	00E3	<^>, <o>	ó	00F3
<^>, <E>	È	00C8	<~>, <O>	Õ	00D5
<^>, <e>	è	00E8	<~>, <o>	õ	00F5
<^>, <E>	É	00C9	<^>, <U>	Û	00D9
<^>, <e>	é	00E9	<^>, <u>	ù	00F9
<~>, <E>	Ê	1EBC	<^>, <U>	Ú	00DA
<~>, <e>	ê	1EBD	<^>, <u>	ú	00FA
<^>, <I>	Ì	00CC	<~>, <U>	Û	0168
<^>, <I>	Í	00CD	<~>, <u>	ü	0169
<~>, <I>	Ĩ	0128	Kombinaciniai diakritiniai ženklai		
<^>, <Y>	Ý	00DD	<^>, <^>	◌̇	0300
<^>, <y>	ý	00FD	<~>, <^>	◌̈	0303
<~>, <Y>	Ỹ	1EF8	<^>, <^>	◌̇̇	0301
<~>, <y>	ỹ	1EF9	<^>, <.>	◌̇̇̇	0307

2 lentelė. Kirčiuotų lietuviškų raidžių įvedimas „Windows“ OS, naudojant tęsties ženklus

pavyzdys taip pat pateiktas 2 lentelėje) – galima įvesti bet kokias kirčiuotas raides. Tačiau šis būdas gan nepatogus ir painus. Beje, nors akivaizdžiai patogesnis ir greitesnis būdas kombinaciniams kirčiams įvesti būtų tiesiog dukart spausti atitinkamą klavišą:

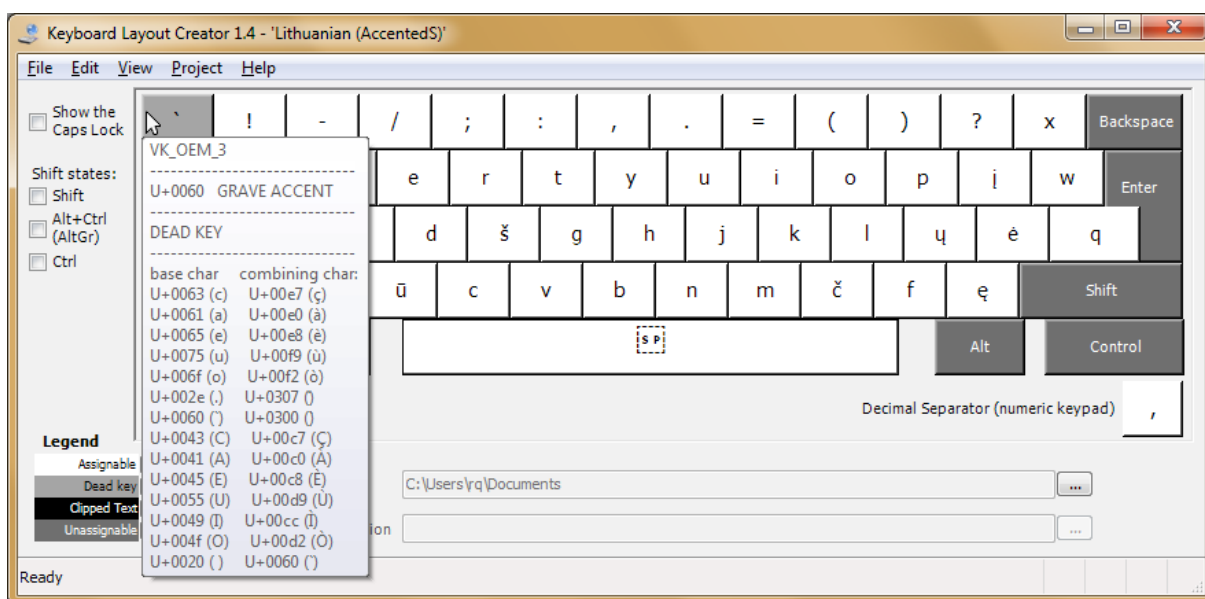
- <^>, <^> → ◌̇,
- <~>, <~> → ◌̈,
- <^>, <^> → ◌̇̇,

jis netinka dirbantiems *Microsoft Word* programa, nes ji kažkodėl „nemėgsta“, kad antruoju ženklu būtų trečiojo lygio ženklas (o dešinysis kirtis būtent toks ir yra). Tokiu atveju jis tiesiog ignoruojamas [Tum07].

Verta pastebėti, jog būna atvejų, kai tiesioginis kombinacinių diakritinių ženklų įvedimas yra tiesiog būtina ir labai patogi savybė (pavyzdžiui kalbotyrininkams), tačiau tai neturėtų būti vienintelis įmanomas būdas dažniau naudojamoms kirčiuotoms raidėms įvesti.

## 2.2. Klaviatūros išdėstymo aprašo kūrimas

Kaip jau minėta, išdėstymų aprašai *Windows* sistemoje turi būti kompiliuojami. Todėl paprasčiausias būdas – juos kurti specialiai tam skirtomis programomis, pavyzdžiui, minėtąja *Microsoft Keyboard Layout Creator* (1 pav.), kurią galima nemokamai parsisiųsti iš *Microsoft* svetainės [Mic09]. Šioje programoje išdėstymo kūrimas vyksta vizualiai, ji labai nesunkiai perprantama, todėl šiame darbe nebus plačiau nagrinėjama.



1 pav. *Microsoft Keyboard Layout Creator* programos vaizdas

Lietuviškos kirčiuotos klaviatūros aprašas, tinkamas *MSKLC* programai naudoti, pateiktas 1 priede. 2 priede pateikta pagrindinės klaviatūros dalies klavišų identifikatorių („skankodų“) *Windows* sistemoje schema.

## 3. „X Window System“ tvarkyklės

*X Window System* (toliau tekste – *X11*) klaviatūros išdėstymų aprašai saugomi grynojo teksto formatu, jie gali paveldėti vienas kito savybes (ši savybė plačiai naudojama net baziniams išdėstymams – pavyzdžiui, visi lietuviški išdėstymai yra aprašyti taip, kad pirma paveldėtų latin išdėstymo savybes, o tada tik pakeičia reikiamų klavišų apibrėžtis), o simbolių kombinavimo (naudojantis tęsties ženklais ar žemiau aprašomu *Compose* klavišu) savybės yra aprašomos nuo paties klaviatūros išdėstymo nepriklausomoje *komponavimo lentelėje* – kaip naudojamos lokalės, o ne išdėstymo dalis. Tai leidžia naudoti ir tobulinti tęsties funkcijas nepriklausomai nuo pasirinkto klaviatūros išdėstymo (ir atvirkščiai). *X11* klaviatūros gali turėti daugiau nei keturis lygius, tačiau ši galimybė beveik neišnaudojama –

šiuo metu penktąjį lygį galima aptikti tik tai daugiakalbiame Kanados išdėstyme bei viename iš vokiškų išdėstymų. Beje, trečiasis ir ketvirtasis lygiai čia plačiai naudojami – bent jau minėtojo *latin* išdėstymo šiuose lygiuose yra patalpinti atitinkami ISO/IEC 9995-3 standarto penktajame skyriuje nurodyti ženklai (tiesa, su keletu nukrypimų).

*X11* atveju labai daug pasirinkimo laisvės turi kompiuterio vartotojas: klaviatūros išdėstymo nuostatų lange jis gali:

- norimiems klavišams pridėti esperanto diakritikus ir/ar euro ženklą,
- pridėti įprasto ir/ar siaurojo jungiamųjų tarpų įvedimo funkciją tarpo klavišui,
- keisti *Lyg2*, *Alt*, *Win*, *Didž* ir kitų funkcinų klavišų veikseną ir/ar poziciją,
- norimam klavišui priskirti *Compose* klavišo funkciją,
- keisti *Lyg3* klavišo poziciją,
- keisti klaviatūros skaitmenų srities veikseną ir savybes,
- priskirti norimam klaviatūros indikatoriumi nepagrindinio naudojamo išdėstymo indikavimo funkciją ir t.t.

Tačiau tai – dar ne viskas. Vartotojas, net ir nebūdamas kompiuterio administratoriumi, gali apsibrėžti nuosavą komponavimo lentelę, ir ši bus iškart naudojama vietoje numatytosios. Be to, naudodamasis standartinėmis *X11* priemonėmis (*xmodmap* programa), jis gali apsirašyti praktiškai bet kokius pakeitimus savo naudojamuose klaviatūros išdėstymuose (ar net susikurti savo išdėstymą) [Kud09].

Panašią laisvę čia turi ir klaviatūros išdėstymo autorius, tik nebūtų tikslinga ja piktnaudžiauti, todėl paprastai klaviatūros išdėstymuose galima rasti panaudotų gan nedaug šių „išplėstinių“ galimybių, o apsiribojama tik tuo, ko tikrai reikia (pavyzdžiui, lietuviškos standartinės klaviatūros apraše yra numatyta, jog trupmenos skyriklis yra kablelis, *Lyg3* funkcija priskirta dešiniajam *Alt* klavišui, o tarpo klavišo trečiuoju ir ketvirtuoju lygiais įvedamas jungiamasis tarpas).

### 3.1. Kirčiuotųjų raidžių įvedimas

Įvesti kirčiuotas lietuviškas raides *X Window System* aplinkoje galima net trimis būdais:

- per tęsties ženklus,

- naudojantis *Compose* klavišu,
- tiesiogiai įvedant kombinacinius diakritinius ženklus.

Visi būdai tarpusavyje yra labai panašūs ir leidžia pasiekti analogiškų rezultatų, tačiau visi šiek tiek ir skiriasi. Bet kuriuo iš šių būdų galima įvesti daugiau nei vieną unikodo kodą, todėl kompozicinių sekų įvedimas iš esmės nėra nesiskiria nuo paprastų raidžių įvedimo (beje, tai įmanoma net ir paprastu bet kurio klavišo paspaudimu, tačiau jau šiek tiek sudėtingiau [Fre09]).

### 3.1.1. Tęsties ženklai

Tęsties ženklų mechanizmas *X11* realizuotas, galima sakyti, idealiai – t.y., jis veikia būtent taip, kaip aprašyta 1.1.1 poskyryje ir parodyta 1 lentelėje.

Prireikus, tęsties ženklus galima kombinuoti tarpusavyje, taip gaunant raidę su daugiau nei vienu kirčiu. Pavyzdžiui, norint įvesti raidę „mažoji a nosinė su riestiniu kirčiu ir dešiniu kirčiu“, tai būtų daroma, paėiliui spaudžiant visus abu tęsties ženklus – dešinį kirtį ir riestinį kirtį, o tada spausti pamatinės raidės klavišą <a>, taigi, seka gautųsi tokia: <’>, <~> <a>, o rezultatas – raidė â (kurią sudaro trijų unikodo simbolių seka: â̂̂̂). Beje, tęsties ženklų įvedimo tvarka yra susitarimo reikalas. Susitarimas yra „iš viršaus į apačią“, tačiau taip pat sėkmingai galima naudoti priešingą tvarką.

Bendruoju atveju įvedimą naudojant tęsties klavišus *X11* aplinkoje galima aprašyti tokiu formalizmu:

$$d_1...d_u c_1 \rightarrow k_1...k_v \quad ,$$

čia  $d_1...d_u$  – bet koks kiekis tęsties ženklų,  $c_1$  – paprastas klaviatūros ženklas,  $k_1...k_v$  – bet koks kiekis atskirus unikodo kodus turinčių simbolių.

### 3.1.2. „Compose“ klavišas

*Compose* klavišas – tai palikimas, paveldėtas iš senųjų darbui su *UNIX* sistemomis skirtų kompiuterių. Šis skirtingų gamintojų komponuotas skirtingose vietose [Wik09] klavišas savo veikimu labai panašus į tęsties ženklą. Bene vieninteliai jo skirtumai nuo įprastų įvedimo klavišų – tai ypatinga padėtis (tarp ženklų srities valdymo klavišų), antrinių funkcijų neturėjimas bei (kai kuriais atvejais) atskiras indikatorius klaviatūroje. Nors šiuolaikinėse

klaviatūrose šis klavišas mažai kur berandamas (turbūt tik *Sun Microsystems* gaminiuose), jo funkciją, kaip, beje, ir bet kurio kito klavišo, nesunkiai galima priskirti bet kuriam klaviatūroje jau esančiam klavišui (pavyzdžiui, vienam iš *X11* aplinkose praktiškai nenaudojamų *Windows* klavišų). Naudojant *Compose* klavišą, kirčiuotos raidės gali būti renkamos, pagal 3 lentelėje pateiktą schemą. Pastaba: lentelėje *Compose* klavišas yra žymimas simboliu  $\text{⌘}$  – būtent toks yra jo žymuo pagal ISO/IEC 9995-7 standartą.

Jei virš raidės reikia uždėti du kirčius ar daugiau, tai nesudaro jokių problemų (visiškai nesvarbu, ar kirčiai kombinaciniai, ar ne):

- $\langle \text{Compose} \rangle, \langle ' \rangle, \langle \sim \rangle, \langle \grave{a} \rangle \rightarrow \acute{\acute{a}}$ .

Atrodytų, jog šis metodas – tai tik pirmojo apsunkinimas (juk tenka spausti vienu klavišu daugiau!), tačiau tai netiesa. Pirmiausia dėl to, kad, naudojant *Compose* klavišą, tęsties ženklų uždelstas veikimas nekliudo darbui, nes jų tiesiog nebereikia – galima palikti įprastus ženklus. *Compose* klavišo paskirtį iš principo galima apibūdinti kaip tęsties ženklų sužadinimą. Iš čia išplaukia ir antras didelis privalumas – jį nuspaudus, tęsties ženklais iš principo tampa visi klaviatūroje esantys ženklai, todėl intuityviai galimų įvesti ženklų aibė žymiai padidėja. Pavyzdžiui:

- $\langle \text{Compose} \rangle, \langle o \rangle, \langle e \rangle \rightarrow \text{œ}$
- $\langle \text{Compose} \rangle, \langle o \rangle, \langle a \rangle \rightarrow \text{å}$
- $\langle \text{Compose} \rangle, \langle t \rangle, \langle m \rangle \rightarrow \text{™}$
- $\langle \text{Compose} \rangle, \langle y \rangle, \langle = \rangle \rightarrow \text{¥}$

ir t. t. Be to, tas pačias kirčiuotas raides įvesti galima ir spaudžiant klavišus priešinga – postfiksine tvarka. Tiesa, kadangi viena seka negali sudaryti kitos sekos pradžios (žr. žemiau), šiuo atveju tektų apsiriboti tik vienu kirčiu virš raidės:

- $\langle \text{Compose} \rangle, \langle i \rangle, \langle ` \rangle \rightarrow \grave{\text{i}}$
- $\langle \text{Compose} \rangle, \langle i \rangle, \langle \sim \rangle \rightarrow \tilde{\text{i}}$
- $\langle \text{Compose} \rangle, \langle i \rangle, \langle ' \rangle \rightarrow \acute{\text{i}}$

ir pan.

Beje, tiek sekų su *Compose* klavišu, tiek tokiomis sekomis išvedamo teksto ilgis nėra ribojamas, tad praktiškai įmanomos net ir sekos, kuriose po *Compose* seka vienas ar dešimt klavišų (arba išvedančios ilgą sakinį vos dviejų klavišų paspaudimu). Vienintelis taikomas

Paeiliui spaudžiami klavišai	Gaunama matoma raidė	Gautos raidės Unikodinė išraiška	Paeiliui spaudžiami klavišai	Gaunama matoma raidė	Gautos raidės Unikodinė išraiška
<⌂>, <`>, <A>	À	00C0	<⌂>, <`>, <Y>	Ý	00DD
<⌂>, <`>, <a>	à	00E0	<⌂>, <`>, <y>	ý	00FD
<⌂>, <´>, <A>	Á	00C1	<⌂>, <~>, <Y>	Ỹ	1EF8
<⌂>, <´>, <a>	á	00E1	<⌂>, <~>, <y>	ỹ	1EF9
<⌂>, <~>, <A>	Ã	00C3	<⌂>, <~>, <J>	Ĵ	004A 0303
<⌂>, <~>, <a>	ã	00E3	<⌂>, <~>, <j>	ĵ	006A 0307 0303
<⌂>, <´>, <Ą>	Ą	0104 0301	<⌂>, <~>, <L>	Ĺ	004C 0303
<⌂>, <´>, <a>	ą	0105 0301	<⌂>, <~>, <l>	ĺ	006C 0303
<⌂>, <~>, <Ą>	Ą	0104 0303	<⌂>, <~>, <M>	Ĺ	004D 0303
<⌂>, <~>, <a>	ą	0105 0303	<⌂>, <~>, <m>	Ĺ	006D 0303
<⌂>, <`>, <E>	È	00C8	<⌂>, <~>, <N>	Ñ	00D1
<⌂>, <`>, <e>	è	00E8	<⌂>, <~>, <n>	ñ	00F1
<⌂>, <´>, <E>	É	00C9	<⌂>, <`>, <O>	Ò	00D2
<⌂>, <´>, <e>	é	00E9	<⌂>, <`>, <o>	ò	00F2
<⌂>, <~>, <E>	Ê	1EBC	<⌂>, <´>, <O>	Ó	00D3
<⌂>, <~>, <e>	ê	1EBD	<⌂>, <´>, <o>	ó	00F3
<⌂>, <´>, <Ę>	Ę	0118 0301	<⌂>, <~>, <O>	Ö	00D5
<⌂>, <´>, <ę>	ę	0119 0301	<⌂>, <~>, <o>	ö	00F5
<⌂>, <~>, <Ę>	Ė	0118 0303	<⌂>, <~>, <R>	Ř	0052 0303
<⌂>, <~>, <ę>	ė	0119 0303	<⌂>, <~>, <r>	ř	0072 0303
<⌂>, <´>, <Ė>	Ė	0116 0301	<⌂>, <`>, <U>	Û	00D9
<⌂>, <´>, <ė>	ė	0117 0301	<⌂>, <`>, <u>	ù	00F9
<⌂>, <~>, <Ė>	Ė	0116 0303	<⌂>, <´>, <U>	Ú	00DA
<⌂>, <~>, <ė>	ė	0117 0303	<⌂>, <´>, <u>	ú	00FA
<⌂>, <`>, <I>	Ì	00CC	<⌂>, <~>, <U>	Ũ	0168
<⌂>, <`>, <i>	ì	0069 0307 0300	<⌂>, <~>, <u>	ũ	0169
<⌂>, <´>, <I>	Í	00CD	<⌂>, <´>, <Ų>	Ų	0172 0301
<⌂>, <´>, <i>	í	0069 0307 0301	<⌂>, <´>, <ų>	ų	0173 0301
<⌂>, <~>, <I>	Ĭ	0128	<⌂>, <~>, <Ų>	Ų	0172 0303
<⌂>, <~>, <i>	ĭ	0069 0307 0303	<⌂>, <~>, <ų>	ų	0173 0303
<⌂>, <´>, <Į>	Į	012E 0301	<⌂>, <´>, <Ū>	Ū	016A 0301
<⌂>, <´>, <į>	į	012F 0307 0301	<⌂>, <´>, <ū>	ú	016B 0301
<⌂>, <~>, <Į>	Ĳ	012E 0303	<⌂>, <~>, <Ū>	Ū	016A 0303
<⌂>, <~>, <į>	į	012F 0307 0303	<⌂>, <~>, <ū>	ũ	016B 0303

3 lentelė: Kirčiuotų lietuviškų raidžių įvedimas, naudojant „Compose“ (⌂) klavišą

apribojimas šioms sekoms – tai kad viena nesudarytų kitos posekio. Tai reiškia, jog jei apibrėžtume dvi sekas, pavyzdžiui, <Compose> <a> <e> <`> bei <Compose> <a> <e>, tai pastarosios rezultato niekada neišvystume ekrane.

### 3.1.3. Kombinacinių diakritinių ženklų įvedimas

Norimų kombinacinių diakritinių ženklų įvedimą galima priskirti tiesiogiai klaviatūros klavišams (arba jų kombinacijai su tęsties ženklu ar *Compose* klavišu). Tokiu atveju juos „uždėti“ būtų galima ant bet kurio tekste jau esančio simbolio. Vis dėlto, šis būdas yra turbūt mažiausiai patrauklus dėl 1.1.2 poskyryje minėtų trūkumų. Kiti du būdai yra žymiai lankstesni.

## 3.2. Klaviatūros išdėstymo aprašo kūrimas

*X11* yra vienintelė tirta aplinka, kurios klaviatūrų išdėstymams kurti nėra specializuotų įrankių. Tai galima paaiškinti skyriaus pradžioje minėtu paveldimumo faktoriumi (dauguma išdėstymų yra „grindžiami“ kitais, pakeisdami tik reikiamų klavišų paibrėžtis. Kitas paaiškinimas – sukurti aprašą paprastu teksto redaktoriumi yra ištis nesunku. Tolesniame pavyzdyje parodoma, kaip tai padaryti (pavyzdys pritaikytas vieną iš *X11* serverio atmainų – *X.org*, naudojamai *Ubuntu Linux* operacinėje sistemoje; kitose sistemose gali šiek tiek skirtis failų keliai).

Tam, kad išdėstymą būtų galima naudoti, jis turi būti atitinkamai sukurtas ir užregistruotas. Lietuviški išdėstymai kuriami faile `/usr/share/X11/xkb/symbols/lt`, kitų kalbų išdėstymai – analogiškai pavadintuose failuose tame pačiame aplanke.

Jeigu šiame faile jau yra lietuviškos standartinės klaviatūros aprašas, tai, norint pridėti kirčiujotąją, tereikia į jo pabaigą (arba tarp kitų išdėstymų aprašų) įterpti tokį teksto bloką:

```
// Lithuanian Standard (LST 1582:2000) keyboard with dead keys.
// for entering accents (stress marks).
partial alphanumeric_keys modifier_keys
xkb_symbols "std_deadkeys" {
    include "lt(std)"
    name[Group1]= "Lithuania - Standard with dead keys";
    key <TLDE> {[ dead_grave, dead_tilde, dead_acute      ]};
    key <AE06> {[ NoSymbol,   NoSymbol, dead_circumflex ]};
    key <AC11> {[ NoSymbol,   NoSymbol, dead_diaeresis  ]};
};
```

Pirmosios dvi šio bloko eilutės yra komentaras. Jis nebūtinai, tačiau papildomai paaiškina failą skaitančiam žmogui, kas yra aprašyta tolesniame bloke.

Trečiają eilutę nurodoma, jog tai yra dalinis išdėstymo aprašas, kuriuo aprašomi tik klaviatūros ženklų srities klavišai (t.y., pagrindinės dalies ir valdymo klavišai).

Ketvirtoji eilutė nurodo vardą, kuriuo šis išdėstymas identifikuojamas (`std_deadkeys`). Riestinis skliaustelis eilutės gale nurodo, jog toliau seka pats aprašas.

Penktąją eilutę į šį aprašą „įtraukiamas“ standartinės lietuviškos klaviatūros išdėstymas, kitaip tariant, nurodoma, jog mūsų kuriamas išdėstymas yra daromas standartinio pagrindu. Šis išdėstymas yra jau aprašytas redaguojamame faile, jis identifikuojamas vardu `std`.

Kitoje eilutėje yra nurodomas jau „gražesnis“ kuriamo išdėstymo pavadinimas.

Eilutės su `key` direktyvomis – tai klavišų apibrėžtys. Kadangi kuriama kirčiuotoji klaviatūra tik trijų klavišų veikimu skiriasi nuo standartinės, būtent šie klavišai čia ir apibrėžti. Klavišo apibrėžties formatas paprastas:

```
key <KLAVIŠO_VARDAS> {[ ženklo_vardas* ]};
```

čia `<KLAVIŠO_VARDAS>` – tai fizinio klavišo identifikatorius, o `ženklų_vardas` – juo įvedamo įprasto arba tęties ženklo vardas. Visi galimi ženklų vardai (tiesa, su pridėtu `XK_` priešdėliu) yra apibrėžti faile `/usr/include/X11/keysymdef.h`. Jei klavišu norima įvesti unikodinį simbolį, kuriam nėra priskirtas ženklo vardas, galima rašyti jo šešiolyktainį unikodo kodą su U raide priekyje. Pavyzdžiui, jei norėtume, kad E00 klavišas įvestų ne tęties ženklus, bet atitinkamus kombinacinius diakritinius ženklus, jį galėtume apibrėžti taip:

```
key <TLDE> {[ U0300, U0303, U0301 ]};
```

Vietoje ženklo vardo galima vartoti ir žodį `NoSymbol` – jis tiesiog paliks galioti ankstesnį aprašą. Tai naudinga, jei norima keisti, pavyzdžiui, trečiojo lygio apibrėžtį, nekeičiant ir naujai neapibrėžiant antrojo ir pirmojo lygių.

Aprašius naują klaviatūros išdėstymą, jį reikia užregistruoti. Tam jį reikia įrašyti į šiuos failus:

```
/usr/share/X11/xkb/rules/base.lst,  
/usr/share/X11/xkb/rules/base.xml,  
/usr/share/X11/xkb/rules/evdev.lst,  
/usr/share/X11/xkb/rules/evdev.xml,  
/usr/share/X11/xkb/symbols.dir.
```

Pradėsime nuo `base.lst` ir `evdev.lst` failų (jie vienodi). Atvėrus šiuos failus teksto redaktoriumi, juose reikia rasti tokią eilutę:



`std`                      `lt: Standard`

ir iškart po jos įterpti štai tokią:

`std_deadkeys`        `lt: Standard with dead keys.`

Įrašius abu failus, galima imtis `base.xml` ir `evdev.xml`. Atvėrus juos teksto redaktoriumi, reikia rasti tokį kodo bloką:

```
<configItem>
  <name>lt</name>
  <shortDescription>Ltu</shortDescription>
  <description>Lithuania</description>
  <languageList><iso639Id>lit</iso639Id></languageList>
</configItem>
<variantList>
  <variant>
    <configItem>
      <name>std</name>
      <description>Standard</description>
    </configItem>
  </variant>
```

Iškart po šio bloko reikia pridėti štai tokias eilutes:

```
<variant>
  <configItem>
    <name>std_deadkeys</name>
    <description>Standard with dead keys</description>
  </configItem>
</variant>
```

Jas pridėjus, ir įrašius, reikia atverti failą `symbols.dir`, jame rasti štai tokią eilutę:

```
--p----- am----- lt(std)
```

ir iškart po jos pridėti štai tokią:

```
--p----- am----- lt(std_deadkeys)
```

Įrašius šį failą, pakanka perleisti `X.org`, ir šią klaviatūrą bus galima išsirinkti savo naudojamose aplinkos klaviatūros nuostatose.

Failas su visų lietuviškų klaviatūrų aprašais (tame tarpe ir aukščiau nagrinėtu klaviatūros su tęsties klavišais aprašu), tinkamas naudoti su *X Window System*, yra pateiktas 3 šio darbo priede. 4 priede pateikta pagrindinės klaviatūros dalies klavišų identifikatorių schema.

### 3.3. Komponavimo lentelė ir jos kūrimas

Visos leidžiamos klavišų, kuriais komponuojamos raidės, sekos yra apibrėžiamos komponavimo lentelėje – vadinamajame **Compose** faile. Nesant šių apibrėžčių, neturėtų prasmės nei tęsties, nei *Compose* klavišų priskyrimai.

Sisteminės komponavimo lentelės yra laikomos katalogo `/usr/share/X11/locale/` pakatalogiuose esančiuose **Compose** failuose. Pakatalogių vardus sudaro lokalės arba koduotės, kurioms šie failai skirti, vardai. Pavyzdžiui, pilnas kelias iki bene didžiausio ir pilniausio sisteminio **Compose** failo yra `/usr/share/X11/locale/en_US.UTF-8/Compose`. Beje, nesant **Compose** failo, skirto lietuviškai lokalei, naudojančiai UTF-8 koduotę, yra naudojamas būtent šis.

Kiekvienas naudotojas turi ir galimybę naudoti savo komponavimo lentelę, įrašydamas ją į failą, kurio numatytasis vardas – `~/XCompose` [Fre06] (ženklą „~“ *UNIX* tipo sistemose žymimas pilnas kelias iki naudotojo namų katalogo). Be to, pasinaudojant `include` direktyva, į šį failą galima įtraukti kitą **Compose** failą, o štai tokia eilutė:

```
include "%L"
```

leis įtraukti savo naudojamos lokalės numatytąjį **Compose** failą. Tai patogu, kai komponavimo lentelėje norima atlikti nedidelius pakeitimus ar papildymus, tačiau tuo pačiu tai reiškia ir šokią tokią riziką, nes sisteminiuose failuose gali pasitaikyti apibrėžčių, kurios konfliktuotų su naudotojo aprašytomis.

Komponavimo lentelės eilučių sintaksė yra apibrėžta jo analizatoriaus pradinio teksto komentare [Fre07]. Mūsų atvejui užteks (išskyrus ką tik minėtą `include` direktyvą) jos supaprastinimo iki tokios formos:

```
<klavišo_reikšmė> * : "Išvedami simboliai" klavišo_reikšmė?
```

Kairėje dvitaškio pusėje yra eilės tvarka išvardijama seka klavišų, kurie bus spaudžiami, o dešinėje – į kabutes įtraukti simboliai (arba vienas simbolis), kurie turi būti gaunami, tokią seką suspaudus. Jei išvedamas tik vienas simbolis, po uždarančios kabutės gali būti užrašomas ir tą simbolį atitinkančio virtualaus klavišo vardas, tačiau tai nebūtina. Apibrėžčių pavyzdžiai:

```
<dead_grave> <I> : "İ" Igrave # I-kairinis
<Multi_key> <I> <grave> : "İ" Igrave # I-kairinis
```

<dead_grave> <i>	:	"i <sup>˙</sup> ˘"	# i-kairinis
<Multi_key> <i> <grave>	:	"i <sup>˙</sup> ˘"	# i-kairinis

5 priede pateiktas komponavimo lentelės failas, naudotinas su šia klaviatūra ir/arba *Compose* klavišu.

#### 4. „Mac OS X“ tvarkyklės

*Mac OS X* (toliau – *OS X*) – tai dešimtoji kompanijos *Apple* kuriamos operacinės sistemos karta. Ši *Mac OS* karta yra *UNIX* tipo operacinė sistema (naujausia jos laida – 10.5 – netgi yra sertifikuota kaip *UNIX*) ir yra, galima sakyti, visiškai kita sistema, nei ankstesnės *Mac OS* kartos (paprastai vadinamos *Mac OS Classic*). Su *OS X* (pradedant 10.2 laida) galima naudoti dviejų tipų klaviatūros išdėstymų aprašus:

1. Senojo *Mac OS Classic* formato dvejetainius resursų failus ir
2. grynojo teksto (*XML*) failus.

Abiejų tipų aprašų galimybės yra identiškos (nors *XML* variantas yra akivaizdžiai patogesnis rankiniam redagavimui, ir tai galima daryti net nenaudojant *OS X*). Be to, abiejų tipų išdėstymus galima „apipavidalinti“, patalpinant juos į specialią katalogų struktūrą. Tai padarius, galima lokalizuoti jų pavadinimus ir priskirti piktogramas [App02]. *XML* tipo aprašus galima kurti ir vizualiai – tam skirta nemokama programa *Ukelele* [SIL09].

*OS X* vartotojas negali konfigūruoti jokių klaviatūros išdėstymo niuansų, tačiau net ir nebūdamas kompiuterio administratoriumi, jis gali įsidiegti naujus išdėstymus – tokiu atveju šie bus matomi ir pasiekiami tik jam pačiam.

*OS X* turi keletą su klaviatūromis ir jų išdėstymais susijusių ypatumų:

1. ši sistema šiek tiek skiriasi nuo kitų tirtų sistemų tradiciniu *Didž* klavišo traktavimu: jo veikimo čia *Lyg2* klavišas „nenumuša“ – net jį ir nuspaudus, toliau bus įvedamos didžiosios raidės;
2. kaip ir *X11*, čia yra plačiai naudojami trečiasis (*Option*) ir ketvirtasis (*Option* + *Lyg2*) lygiai, tik čia juose panašu, jog patalpinti beveik atsitiktiniai simboliai;
3. Apple kompiuterių amerikinėse klaviatūrose klavišo B00 kodas yra toks pat, kaip klavišo E00 kodas europinėse. Kitaip tariant, šis klavišas „peršoka“ iš vienos pozicijos į kitą. Šio fakto reikia nepamiršti, kuriant išdėstymą, kurį ketinama platinti. Jei

norima, kad sukurtu sukurtuoju išdėstymu galėtų patogiai naudotis ir amerikinių klaviatūrų turėtojai, reikia arba kituose klavišuose dubliuoti europinės klaviatūros E00 klavišo funkcijas, arba kurti modifikuotą išdėstymo variantą amerikinėms klaviatūroms.

Atskirą klaviatūros lygį *OS X* galima apibrėžti pagal tokius požymius:

1. ar nuspaustas kairysis *Lyg2* klavišas,
2. ar nuspaustas dešinysis *Lyg2* klavišas,
3. ar nuspaustas bet kuris *Lyg2* klavišas,
4. ar nuspaustas kairysis *Option* klavišas,
5. ar nuspaustas dešinysis *Option* klavišas,
6. ar nuspaustas bet kuris *Option* klavišas,
7. ar nuspaustas kairysis *Vald* klavišas,
8. ar nuspaustas dešinysis *Vald* klavišas,
9. ar nuspaustas bet kuris *Vald* klavišas,
10. ar nuspaustas *Command* klavišas,
11. ar aktyvintas *Didž* klavišas.

Visi šie požymiai yra trejetainiai („taip“, „ne“, „nesvarbu“). Taigi, atmetus kombinacinius požymius (3-ąjį, 6-ąjį ir 9-ąjį) bei požymių variantus („nesvarbu“), lieka net  $2^8 = 256$  teoriškai įmanomi diskretūs klaviatūros lygiai. Vis dėlto, atmetus nepatartinus naudoti ir/ar tiesiog nepraktiškus variantus, jų lieka mažiau. O praktikoje dažniausiai naudojami šie lygiai [SIL09]:

1. be modifikatorių;
2. nuspaustas *Lyg2* klavišas, *Didž* būseną nesvarbi;
3. nuspaustas *Option* klavišas;
4. nuspausti *Lyg2* ir *Option* klavišai;
5. aktyvintas *Didž* klavišas;
6. nuspaustas *Command* klavišas, *Lyg2* būseną nesvarbi;

7. nuspaustas *Vald* klavišas, kitų klavišų būsenos nesvarbios.

Pastarieji du lygiai taip pat neturėtų būti naudojami simboliams įvesti. Klavišų kombinacijas su *Command* programos interpretuoja kaip komandas (kaip *Windows* ir *X11* atveju interpretuojamos klavišų kombinacijos su *Vald*), todėl šiame lygyje kartais specialiai nurodomas amerikinis *QWERTY* išdėstymas – tam, kad spartieji klavišai veiktų taip pat, kaip jame. Tuo tarpu kombinacijų su *Vald* klavišu aprašus *OS X* apskritai ignoruoja, visada naudodama numatytuosius (taip teigia *Ukelele* instrukcija, kurioje pateikiamas ir paaiškinimas, kaip šią problemą apeiti). Tiesa, lygių su nuspaustais *Vald* ir *Alt* klavišais aprašai nėra ignoruojami. Vis dėlto, reikia nepamiršti, jog klaviatūros išdėstymas turėtų būti dar ir intuityvus, todėl piktnaudžiauti lygių skaičiumi neverta – priešingu atveju išdėstymas taptų tiesiog pernelyg sudėtingas.

#### 4.1. Kirčiuotųjų raidžių įvedimas

*OS X* sistemoje galima numatyti du būdus kirčiuotoms raidėms įvesti:

- per tęsties ženklus,
- tiesiogiai įvedant kombinacinius diakritinius ženklus.

Tiek paprastu vienu klavišo paspaudimu, tiek per tęsties ženklus galima įvesti daugiau nei vieną unikodo kodą, todėl kompozicinių sekų įvedimas nėra kiek nesiskiria nuo turinčių unikodo kodus raidžių įvedimo.

##### 4.1.1. Tęsties ženklai

Tęsties ženklų mechanizmas *OS X* atveju realizuotas puikiai. Jis ne tik veikia taip, kaip aprašyta 1.1.1 poskyryje ir pavaizduota pirmoje lentelėje, tačiau turi ir dar vieną labai malonų privalumą – savo esme būtent tokį, kokį turėjo spausdinimo mašinėlės – tai vizualų tęsties ženklo indikavimą. Nuspaudus šį ženklą, jis iškart parodomas ekrane, tačiau kitos spalvos, nei aplink esantis tekstas, fone (2 pav.).

Prireikus, tęsties ženklus galima kombinuoti tarpusavyje, taip gaunant raidę su daugiau nei vienu kirčiu. Pavyzdžiui, norint įvesti raidę „mažoji a nosinė su riestiniu kirčiu ir dešiniu kirčiu“, tai būtų daroma, paeiliui spaudžiant visus abu tęsties ženklus – dešinini kirtį ir riestinį kirtį, o tada spausti pamatinės raidės klavišą <a>, taigi, seka gautųsi tokia: <^>, <~> <a>, o rezultatas – raidė á (kurią sudaro trijų unikodo simbolių seka: á̂̃̄). Tęsties

ženklų įvedimo tvarka yra susitarimo reikalas – galima sukurti klaviatūros išdėstymo aprašą, kuriame kirčiai bus kombinuojami ne iš viršaus į apačią, bet iš apačios į viršų. Tiesa, jis būtų neintuityvus.

Bendruoju atveju įvedimą naudojant tęties klavišus *Mac OS X* aplinkoje galima aprašyti tokiu formalizmu:

$$d_1...d_u c_1 \rightarrow k_1...k_v ,$$

čia  $d_1...d_u$  – bent vienas tęties ženklas,  $c_1$  – paprastas klaviatūros ženklas,  $k_1...k_v$  – bet koks kiekis atskirus unikodo kodus turinčių simbolių.



2 pav. Tęties ženklo indikavimas  
„Mac OS X“ aplinkoje

#### 4.1.2. Kombinacinių diakritinių ženklų įvedimas

Norimų kombinacinių diakritinių ženklų įvedimą galima priskirti tiesiogiai klaviatūros klavišams (arba jų kombinacijoms su tęties ženklais). Tokiu atveju juos „uždėti“ būtų galima ant bet kurio tekste jau esančio simbolio. Vis dėlto, šis būdas yra turbūt mažiausiai patrauklus dėl 1.1.2 poskyryje minėtų trūkumų.

#### 4.2. Klaviatūros išdėstymo aprašo struktūra ir kūrimas

*Mac OS X*, kaip ir *Windows* atveju, turbūt patogiausias išdėstymo kūrimo būdas yra daryti tai vizualiai – pasinaudojant jau minėta nemokama programa *Ukelele* [SIL09].

Beje, priešingai nei *MSKLC* atveju, ši programa nenaudoja atskiro formato redaguojamam išdėstymui saugoti – ji operuoja tais pačiais *XML* tipo klaviatūros išdėstymų aprašais, kaip ir pati operacinė sistema. Kaip ir *MSKLC*, ši programa yra pakankamai



3 pav.: Programos „Ukelele“ pagrindinis langas

intuityvi (3 pav.) ir turi gan išsamų žinyną, todėl ją aptarinėti šiame darbe būtų neprasminga. Kita vertus, jos galimybės šiek tiek ribotos. Todėl šiame skyriuje bus aprašyta, kaip sukurti klaviatūros išdėstymo aprašą (arba tiesiog jį patobulinti) įprastu teksto redaktoriumi.

Pradžią pradžia: klaviatūros išdėstymo *XML* failas turėtų būti įrašomas UTF-8 arba UTF-16 koduote. Jis visada pradedamas standartine antrašte:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE keyboard SYSTEM
"file://localhost/System/Library/DTDs/KeyboardLayout.dtd">
```

Šakninis klaviatūros aprašo elementas – `<keyboard>`. Jis privalo turėti šiuos atributus:

- **group** – rašmenų sistemos, kuri kuriamame išdėstyme naudojama daugiausiai, kodas *OS X* sistemoje. Unikodo grupės kodas – 126, Centrinės Europos grupės – 29. Daugiau informacijos apie šias grupes galima rasti [App02];

- **id** – unikalus klaviatūros identifikatorius. Jo reikšmė turi būti skaitinė. Jei **group** atributas nurodo unikodo grupę, **id** turi būti neigiamas. Jei kelių išdėstymų **id** sutampa, operacinė sistema juos perskirsto savarankiškai;

- **name** – klaviatūros išdėstymo pavadinimas.

Taip pat yra vienas neprivalomas atributas:

- **maxout** – maksimalus unikodo ženklų kiekis, kurį galima gauti vienu klavišo paspaudimu.

`<keyboard>` elementą atidarantios gairės pavyzdys:

```
<keyboard group="126" id="-3806" name="Lithuanian - Accented"
maxout="3">
```

Šiame elemente privalo būti:

- lygiai vienas `<layouts>` elementas,
- bent vienas `<modifierMap>` elementas,
- bent vienas `<keyMapSet>` elementas;

taip pat jame gali būti:

- ne daugiau nei vienas `<actions>` elementas,
- ne daugiau vienas `<terminators>` elementas.

`<layouts>` elemente privalo būti bent vienas elementas `<layout>`. Šiais elementais fizinės klaviatūros susiejamos su kuriama išdėstymo variantais. Skirtingi variantai gali būti naudingi, kai norima pritaikyti išdėstymą kelių tipų fizinėms klaviatūroms. Tai ypač praverstų sprendžiant vieno klavišo kodo „persökimo“ iš vienos amerikinėje *Apple*

klaviatūroje į kitą vietą europinėje problemą. Rašant šį darbą, nepavyko rasti pakankamai šviežio ir išsamaus fizinių klaviatūrų identifikatorių sąrašo, tačiau pavyko jį iš dalies sudaryti (4 lentelė).

Tipas	Klaviatūrų identifikatoriai
ANSI (amerikinė)	1, 2, 3, 6, 8, 10, 11, 12, 16, 24, 27, 28, 31, 34, 37, 40, 43, 46, 195, 198, 202, 204, 259, 1202
ISO (europinė)	4, 5, 7, 9, 13, 17, 20, 29, 32, 35, 38, 41, 44, 47, 196, 199, 203, 205, 1203
JIS (japoninė)	18, 21, 22, 23, 30, 33, 36, 39, 42, 45, 48, 194, 197, 200, 201, 206, 207, 1207

4 lentelė: Fizinių klaviatūrų identifikatoriai Mac OS X sistemoje

`<layout>` elementą galima įsivaizduoti kaip tam tikrą filtrą, pagal klaviatūros identifikatorių jai priskiriantį išdėstymo variantą. Jis privalo turėti šiuo atributus:

- `first` – žemiausias filtrą tenkinantis klaviatūros identifikatorius,
- `last` – aukščiausias filtrą tenkinantis identifikatorius,
- `modifiers` – filtrą tenkinančioms klaviatūroms priskiriamo `<modifierMap>` elemento (apie jį – žemiau) identifikatorius (`id` atributo reikšmė),
- `mapSet` – filtrą tenkinančioms klaviatūroms priskiriamo išdėstymo varianto – `<keyMapSet>` elemento (apie jį – žemiau) identifikatorius.

`<layouts>` elemento aprašo pavyzdys:

```
<layouts>
```

```
  <layout first="0" last="3" modifiers="commonModifiers" mapSet="ANSI" />
```

```
</layouts>
```

Klaviatūroms, kurių identifikatoriai neatitinka nė vieno filtro, bus priskiriami pirmajame filtre nurodyti `<modifierMap>` ir `<mapSet>` elementai.

`<modifierMap>` elementas susieja modifikavimo klavišų būsenų kombinacijas su konkrečiais klaviatūros lygiais. Jame privalo būti vienas arba daugiau `<keyMapSelect>` elementų.

`<modifierMap>` elementui privalomi du atributai:

- `id` – jo identifikatorius, naudojamas `<layout>` elementuose,
- `defaultIndex` – lygis, kuris bus naudojamas tuo atveju, kai modifikavimo klavišų būsenų kombinacija netenkina nė vieno išreikšto filtro (`<modifier>` elemento).

Kiekvienas `<keyMapSelect>` elementas atitinka vieną klaviatūros lygį (`<keyMap>`



elementą, apie jį žemiau). Šiame elemente yra išvardijami modifikavimo klavišų būsenų kombinacijų filtrai (<modifier> elementai), jei bent vieną iš kurių tenkinant, atitinkamas lygis laikomas aktyviu. Jei tas pats filtras būna panaudotas keliuose <keyMapSelect> elementuose, jis veiks tik paskutiniame. <keyMapSelect> elementui privalomas vienas atributas:

- **mapIndex** – nurodo klaviatūros lygį, kurį reikia naudoti, jei tenkinamas bent vienas kombinacijų filtras. Klaviatūros lygiai turi būti numeruojami nuo nulio, kiekvieno tolesnio lygio indeksą didinant vienetu.

Vienas <modifier> elementas nurodo vieną modifikavimo klavišų būsenų kombinaciją, kitaip tariant, filtrą. Jam yra privalomas vienas atributas:

- **keys**. Šis atributas nurodo, kokie modifikavimo klavišai privalo, gali, arba neturi būti paspausti, kad filtras būtų tenkinamas. Šio atributo formatas – paprasta teksto eilutė, kurioje gali būti šie tarpais atskirti žodžiai (gali nebūti ir nė vieno):
  - ◆ **shift** – žymi kairįjį *Lyg2* klavišą,
  - ◆ **rightShift** – žymi dešinįjį *Lyg2* klavišą,
  - ◆ **anyShift** – žymi bet kurį *Lyg2* klavišą,
  - ◆ **option** – žymi kairįjį *Option* klavišą,
  - ◆ **rightOption** – žymi dešinįjį *Option* klavišą,
  - ◆ **anyOption** – žymi bet kurį *Option* klavišą,
  - ◆ **control** – žymi kairįjį *Vald* klavišą,
  - ◆ **rightControl** – žymi dešinįjį *Vald* klavišą,
  - ◆ **command** – žymi *Command* klavišą,
  - ◆ **caps** – žymi *Didž* klavišą.

šių žodžių buvimas eilutėje reiškia, jog atitinkamas klavišas (ar bent vienas iš atitinkamų klavišų) turi būti nuspaustas. Jei žodžio eilutėje nėra, tai reiškia, kad nė vienas juo žymimas klavišas negali būti nuspaustas. Jei eilutėje yra žodis su klaustuku (pvz., **anyOption?**), tai reiškia, jog juo žymimo klavišo (ar klavišų) būseną neįtakoja šio filtro veikimo – klavišas gali būti nuspaustas, o gali būti nenuspaustas.

Pastaba: paprastai OS X į kairiojo ir dešiniojo modifikavimo klavišų nuspaudimą reaguoja vienodai – lyg būtų paspaustas kairysis. Todėl nėra prasmės (tai net gali pakenkti išdėstymo funkcionalumui) aprašinėti kairiojo ir dešiniojo klavišų atskirai – lygio parinkimui geriausia naudoti juos abu žymintį žodį su „any“ priešdėliu (pvz., **anyShift**).

Pilno `<modifierMap>` elemento pavyzdys:

```
<modifierMap id="commonModifiers" defaultIndex="0">
  <keyMapSelect mapIndex="0">
    <!-- modifikatoriai nenuspausti -->
    <modifier keys="" />
  </keyMapSelect>
  <keyMapSelect mapIndex="1">
    <!-- bet kuris Lyg2 nuspaustas, Didž būseną nesvarbi -->
    <modifier keys="anyShift caps?" />
  </keyMapSelect>
  <keyMapSelect mapIndex="2">
    <!-- aktyvintas Didž -->
    <modifier keys="caps" />
  </keyMapSelect>
  <keyMapSelect mapIndex="3">
    <!-- nuspaustas kairysis arba dešinysis Option,
           Didž būseną nesvarbi -->
    <modifier keys="option caps?" />
    <modifier keys="rightOption caps?" />
  </keyMapSelect>
</modifierMap>
```

Pastaba: pavyzdyje specialiai kaip du atskiri filtrai panaudotos eilutės su `option` ir `rightOption` žodžiais. Kadangi jie abu aktyvina tą patį lygį, logiškiau būtų vietoje jų rašyti vieną filtrą: `<modifier keys="anyOption caps?" />`.

`<keyMapSet>` elementui talpinamas vieno išdėstymo varianto aprašas. Paprastai išdėstymuose būna aprašomas vienas arba du variantai. Pastaruoju atveju vienas taikomas europinėms ir amerikinėms klaviatūroms (*ISO* ir *ANSI*), o kitas – japoninėms (*JIS*). Tačiau norint, *ISO* ir *ANSI* taip pat galima išskirti į du variantus.

`<keyMapSet>` elementui privalomas vienas atributas:

- `id` – tai šio išdėstymo varianto identifikatorius. Savaime aišku, jog tas pats išdėstymas negali turėti daugiau nei vieno varianto su tokiu pačiu identifikatoriumi. Geras (tačiau neprivalomas) pavyzdys – vadinti išdėstymą pagal klaviatūros tipą, kuriai jis skirtas (pavyzdys: `<keyMapSet id="ANSI">`).

`<keyMapSet>` elementui privalo būti bent vienas `<keyMap>` elementas. Jis atitinka vieną klaviatūros išdėstymo lygį. `<keyMap>` elementui privalo turėti vieną atributą:

- `index` – šis skaičius nurodo lygio indeksą (numerį). Kaip jau minėta, lygiai

numeruojami nuo nulio, kiekvieno tolesnio lygio indeksą didinant vienetu. Būtent šį indeksą nurodo `<keyMapSelect>` elemento `mapIndex` atributas.

Taip pat šis elementas gali turėti du neprivalomus atributus, reikalingus paveldimumui. Šie atributai yra priklausomi vienas nuo kito ir negali būti naudojami pavieniui. Tai:

- `baseMapSet` – atributas, nurodantis bazinį išdėstymo variantą ir
- `baseIndex` – atributas, nurodantis bazinį bazinio išdėstymo varianto lygį.

Nurodžius šiuos atributus, visi šiame lygyje neapibrėžti klavišai paveldės savo funkcijas iš bazinio išdėstymo varianto bazinio lygio. Ši savybė labai patogi nedideliems pakeitimams, priklausomiems nuo fizinės klaviatūros, atlikti. Pavyzdys:

```
<keyMapSet id="ANSI">
  <keyMap index="0" baseMapSet="ISO" baseIndex="0">
    <!-- suderinamumo pataisos amerikinėms klaviatūroms -->
```

Nenurodžius paveldimumo atributų, lygyje neapibrėžti klavišai bus neveiksnūs – tiesiog nieko nedarys. `<keyMapSet>` elemente privalomas bent vienas `<key>` elementas.

`<key>` elementas aprašo klavišo veikimą konkrečiame lygyje. Šį elementą galima užrašyti trimis būdais:

1. `<key code="klavišo_kodas" output="eilutė" />`

Toks aprašas nurodo, jog nuspaudus klavišą su atitinkamu identifikatoriumi (atributo `code` reikšme), tiesiog bus grąžinta teksto eilutė (atributo `output` reikšmė).

Pavyzdžiui: `<key code="0" output="Labas!" />;`

2. `<key code="klavišo_kodas" action="veiksmas" />`

Toks aprašas nurodo, jog nuspaudus klavišą su atitinkamu identifikatoriumi, bus atliktas veiksmas, kurio identifikatorius nurodytas `action` attribute (apie veiksmus kalbama žemiau). Pavyzdžiui: `<key code="o" action="kairinis" />;`

3. `<key code="klavišo_kodas">`

```
  <action>
    <!-- bevardis veiksmas -->
  </action>
</key>
```

Toks aprašas nurodo, jog paspaudus klavišą su atitinkamu identifikatoriumi, bus atliktas `<action>` elemente aprašytas veiksmas. Nuo antrojo varianto šis skiriasi tik tuo, jog taip aprašyto veiksmo negalima iškviesti iš kitos klaviatūros aprašo vietos.

`<actions>` elementas yra neprivalomas. Jame aprašomi vardiniai veiksmai. Jis neturi privalomų atributų, tačiau jame turėtų būti bent vienas `<action>` elementas.

`<action>` elementu aprašomas vienas veiksmas. Reikia pažymėti, jog *OS X* klaviatūros tvarkyklė veikia iš dalies kaip būsenų automatas. `<action>` elementu yra keičiama jos būsena, ir gali būti išvedamas tekstas. Ką būtent daryti ir kuriuo atveju, nurodoma `<when>` elementais.

`<action>` privalo turėti vieną atributą:

`id` – šio veiksmo identifikatorius. Du ar daugiau veiksmų negali turėti vienodų identifikatorių.

`<when>` elementu apibrėžiama, ką atlieka konkretus veiksmas esant konkrečiai būsenai (arba vienai iš būsenų). Šis elementas gali būti dviejų pavidalų:

1. `<when state="identifikatorius" [next="identifikatorius"] [output="identifikatorius"] />`

Šis būdas nurodo tik vieną įėjties būseną, kurios identifikatorius (atributo `state` reikšmė) gali būti bet koks tekstinis būsenos pavadinimas, jos numeris, arba specialus vardas `none`, reiškiantis pradinę automato būseną.

Neprivalomu atributu `next` galima nurodyti identifikatorių būsenos, į kurią automatas turi pereiti, atlikęs šį veiksmą.

Neprivalomu atributu `output` galima nurodyti, šiuo veiksmu išvedamą tekstą.

2. `<when state="nuo" through="iki" [multiplier="daugiklis"] [next="tolesnė"] [output="simbolis"] />`

Šis būdas nurodo, jog jei įėjties būsenos numeris yra skaičius nuo parametro `state` reikšmės iki parametro `through` reikšmės imtinai (šį numerį žymėsime kintamuoju `būsena`), tai atlikęs veiksmą, automatas turi pereiti į būseną, kurios numeris yra  $(būsena - nuo) * daugiklis + tolesnė$  ir/arba išvesti simbolį, kurio unikodo kodas yra  $(būsena - nuo) * daugiklis + simbolis$ . Šiuo atveju visos būsenos turi būti nurodomo skaičiais, o neprivalomi yra parametrai `multiplier`, `next` ir `output`.

Abiem atvejais nenurodžius būsenos, į kurią reikia pereiti, bus grįžtama į pradinę automato būseną (`none`), o nenurodžius išvedamo teksto, nebus išvedamas joks. Be to, antruoju atveju nenurodžius daugiklio, bus laikoma, jog jis lygus vienetui.

Pastebėtina, jog net ir vardinėms būsenoms, kompiliuojant išdėstymą, suteikiami numeriai. Tačiau kompiliatorius pasirūpina, kad jie būtų didesni už visus apraše panaudotus skaitinius būsenų identifikatorius, todėl vardinių būsenų nepavyks iškviešti, nurodant jų numerį.

Neprivalomame `<terminators>` elemente nurodomi „kraštutiniai“ atvejai. Jei klavišo

paspaudimu yra nurodyta atlikti veiksmą, tačiau veiksmė nepavyko rasti `<when>` elemento, kurį tenkintų esamoji būsenų automato būsena, tuomet vykdomas automata „terminuojantis“ veiksmas, aprašytas `<terminators>` elemente. Šis elementas neturi jokių atributų, tačiau jame privalo būti bent vienas `<when>` elementas. Šiuo atveju `<when>` elementai gali turėti tik du atributus – tai `state` ir `output`. Atlikus terminalinį veiksmą, būsenų automatas grąžinamas į pradinę (`none`) būseną – net ir tuo atveju, jei šiame veiksmė nebuvo rastas nė vienas tinkantis `<when>` elementas. Pavyzdys:

```
<terminators>
  <when state="grave" output="`"/>
  <when state="acute" output="&#x00b4;"/><!-- ACUTE ACCENT -->
  <when state="tilde" output="~"/><!-- TILDE -->
  <when state="circumflex" output="^"/><!-- CIRCUMFLEX -->
  <when state="diaeresis" output="&#x0022;"/><!-- QUOTATION MARK -->
</terminators>
```

6 priede pateiktas su *Mac OS X* naudotinas lietuviškos kirčiuotos klaviatūros aprašas. 7 priede pateikiamas klaviatūros aprašo *DTD (Document Type Definition)* failas. *Apple* klaviatūrų (europinės ir amerikinės) pagrindinės dalies klavišų identifikatorių schemos pateiktos 8 priede.

## Išvados ir pasiūlymai

Palyginus trijose aplinkose – *Windows*, *X Window System* ir *Mac OS X* – naudojamas klaviatūros tvarkykles (tiksliau klaviatūros išdėstymų aprašus), peršasi akivaizdi išvada, jog pastarosios dvi aplinkos galimybėmis *Windows* gerokai lenkia. Jos abi leidžia pasinaudoti tęsties ženklais, kad įvesti kombinacines (ar šiaip bet kokias) ženklų sekas, tuo tarpu *Windows* atveju sekos su tęsties ženklu (ar ženklais) rezultatas visada turi būti vienas vienintelis unikodo kodas.

Šis (atrodytų nežymus) skirtumas tampa esmine kliūtimi, kai norima klaviatūra patogiai įvesti lietuviškas kirčiuotas raides, kurių tik mažiau nei pusė turi priskirtus unikodo kodus, o kitos yra išreiškiamos kompozicinėmis sekomis, kurias sudaro keli unikodo ženklai – *Windows* atveju kai kurių raidžių taip įvesti tiesiog neįmanoma. Su *X Window System* ir *Mac OS X* įmanomas toks įvedimo formalizavimas:

$$d_1 \dots d_u c_1 \rightarrow k_1 \dots k_v \quad ,$$

tuo tarpu su *Windows* – tik toks:

$$d_1 c_1 \rightarrow k_1 \quad ,$$

čia  $d_i$  – tęsties ženklas,  $c_1$  – paprastas klaviatūros ženklas,  $k_i$  – atskirą unikodo kodą turintis simbolis.

*X Window System* ir *Mac OS X* galimybės daugmaž panašios, tačiau abi aplinkos turi savų privalumų: pirmoji leidžia naudotis *Compose* klavišu ir, jei tik norima, visiškai atsisakyti tęsties ženklų (beje, tai įmanoma net ir tada, kai fizinio *Compose* klavišo klaviatūroje nėra – priešingu atveju šis privalumas būtų tiesiog nepastebimas), o antroji gan paprastai, bet tuo pačiu labai intuityviai, vizualizuoja sekų su tęsties ženklais įvedimą.

Darbo prieduose pateiktos lietuviškų klaviatūrų su kirčiuotomis raidėmis aprašai visoms trimis tirtoms aplinkoms (*Windows* atveju pateiktas tokio aprašo pradinis tekstas, kurį norint naudoti, reikia papildomai apdoroti), o taip pat klavišų identifikavimo schemas, kurių gali prireikti, norint pataisyti ar papildyti jau sukurtus išdėstymus grynajo teksto redaktoriui.

## **Analysis of Keyboard Layouts and Drivers** (Summary in English)

Ever since it was invented, the computer keyboard has been the main device used for supplying textual information for computer. However, the keyboard itself does not transmit this (textual) information: instead, it sends the computer information about key presses, and this information is then processed by special software – the keyboard driver. This study is an attempt to analyze the possibilities of these drivers among different popular operating systems: Windows, Mac OS X, and Linux, the latter being represented by the most popular graphics subsystem it uses – the X Window System (X.org in particular).

It's obvious that basic functionality even among different environments is the same. Hence, the study pays most attention to a less often use case: entering characters with accents (stress marks), especially those that don't have Unicode positions for their precomposed forms, and have to be composed using composite sequences consisting from the base character and necessary combining accents.

Main motivation for this study was the fact that standard modern Lithuanian uses three accent (stress) signs: grave (`), acute(´), and circumflex (~). There are 68 total (34 lower, and 34 upper case) Lithuanian accented letters, only 33 of them having Unicode codes. Others must be represented using composite sequences.

This paper reveals a big deficiency in Windows keyboard driver (impossibility to use dead keys to enter more than one Unicode code point) and suggests a way to overcome it. Also, it introduces the ways to create keyboard layout descriptions (drivers) for all three environments and provides such descriptions for the Lithuanian Accented keyboard. Additionally, visual lists of key identifiers in all three environments are provided.

## Literatūros sąrašas

- [App02] Apple Inc. Technical Note TN2056. Installable Keyboard Layouts.  
<http://developer.apple.com/technotes/tn2002/tn2056.html>. 51KB.
- [FR02] J. Fulton, D. Rosenthal. XMODMAP(1) manual page.  
<http://www.x.org/archive/X11R6.8.1/doc/xmodmap.1.html#sect4>. 13 KB.
- [Fre06] Release Notes for X11R7.1: Overview of X11R7.1.  
<http://xorg.freedesktop.org/releases/X11R7.1/doc/RELNOTES4.html#30>. 20 KB.
- [Fre07] xorg/lib/libX11 - libX11 GIT Repository.  
<http://cgит.freedesktop.org/xorg/lib/libX11/tree/modules/im/ximcp/imLcPrs.c?h=xge&id=4ba091255bb953d53078ba5619d6751052c739f7#n59>. 87 KB
- [Fre09] X.org Bug 19506. Support of Breton C'HWERTY keyboard (C'H trigraph and CH digraph). [https://bugs.freedesktop.org/show\\_bug.cgi?id=19506](https://bugs.freedesktop.org/show_bug.cgi?id=19506). 27KB.
- [Kap05] M. Kaplan. It is not easy to chain dead keys on Windows.  
<http://blogs.msdn.com/michkap/archive/2005/11/15/492200.aspx>. 15KB.
- [Kap07] M. Kaplan. MSKLC keyboard layout names in your own language.  
<http://blogs.msdn.com/michkap/archive/2007/08/25/4564548.aspx>. 55KB.
- [Kud09] R. Kudelis. Klaviatūra įvedamų ženklų aibės plėtimo metodai „X Window System“ („X11“) grafinėje aplinkoje. Kursinis projektas, 2009, p. 9-15.
- [Mic09] Microsoft Corporation. The Microsoft Keyboard Layout Creator.  
[http://msdn.microsoft.com/lt-lt/global/bb964665\(en-us\).aspx](http://msdn.microsoft.com/lt-lt/global/bb964665(en-us).aspx). 6KB.
- [SIL09] SIL International. Ukelele. Mac OS X Keyboard Layout Editor.  
<http://scripts.sil.org/Ukelele>. 81KB.
- [Tum07] V. Tumasonis. Lietuviškos kirčiuotos raidės: kodavimas ir įvedimas iš klaviatūros. Informacijos mokslai, 42-43, 2007, p. 135-140.
- [Wik09] Wikipedia, the free encyclopedia. Compose key.  
[http://en.wikipedia.org/wiki/Compose\\_key](http://en.wikipedia.org/wiki/Compose_key). 65 KB.