

# "ScratchJr" metodinė medžiaga

# "ScratchJr" metodinė medžiaga

turinys

"ScratchJr" aplinka ir programavimo kalba Reikalavimai sistemai Veikėjai ir veiksmų laukas "ScratchJr" blokų apžvalga Veikėjų judesių programavimas. Programos paleidimas Garsai Nuoseklusis ir lygiagretusis vykdymas Veiksmų kartojimas Sąlygos ir jvykiai Veikėjų sąveika Susidūrimo įvykis Laukimas Pranešimų siuntimas Puslapiai. Veiksmų lauko keitimas Naujo puslapio kūrimas Perėjimas tarp puslapių Veikėjų ir jų scenarijų kopijavimas

#### "ScratchJr" aplinka ir programavimo kalba

Programavimo priemonė "ScratchJr" sukurta specialiai jaunesniojo amžiaus vaikams, kad pradinių klasių mokiniai ir net ikimokyklinio amžiaus vaikai (autoriai nurodo amžių nuo penkerių metų) galėtų kurti nesudėtingas programas, interaktyvius pasakojimus, žaidimus. "ScratchJr" programavimo kalba yra vaizdinė: programavimo sakiniai dėliojami iš spalvingų blokų.



Naudodamiesi "ScratchJr" priemone, vaikai išmoksta programavimo, planavimo, kompiuterio valdymo pradmenų, mokosi spręsti įvairius uždavinius, kūrybiškai išreikšti save. Priemonėje esantys veikėjai ir fonai skatina vaikus kurti pasakojimus, animacijas, o integruota piešimo priemonė leidžia piešti naujus veikėjus, fonus ar koreguoti esamus. "ScratchJr" programavimo kalba pritaikyta jaunesniojo amžiaus vaikams: programavimo konstrukcijos reiškiamos paveikslėliais, todėl net dar neišmokę rašyti vaikai gali kurti nesudėtingas programėles. Integruota garso įrašymo priemonė leidžia vaikams įrašyti savo balsą ir taip įgarsinti projekto veikėjų pokalbius. Vaikai, gebantys rašyti tekstus, gali kurti sudėtingesnius pasakojimus, interaktyvius tekstinius dialogus. Sukurtus darbelius vaikai gali įrašyti planšetiniame kompiuteryje arba jais pasidalyti su kitais (pavyzdžiui, išsiųsti el. paštu tiesiogiai iš "ScratchJr" aplinkos). Priemonė teikia labai plačias galimybes integruoti informatikos konceptus į įvairius pradinio ugdymo dalykus.

Pradėdami kalbėti apie "ScratchJr" aplinką ir darbo joje pradžią, pirmiausia turime išsiaiškinti šias sąvokas:

#### programavimo kalba

Žymenų sistema kompiuteriu atliekamų veiksmų aprašui pateikti.

#### piktograminė programavimo kalba grafinė programavimo kalba

Programavimo kalba, kurios visos pagrindinės programavimo sąvokos pateikiamos grafiškai, t. y. piktogramomis, blokais.

#### komanda

Daugelyje programavimo kalbų vartojamas nurodymas kompiuteriui atlikti kurį nors veiksmą.

"ScratchJr" aplinkoje vaikai nurodymą (komandą) pateikia ne tiesiogiai kompiuteriui, o **veikėjui**, kuris tą komandą atlieka. Komandos reiškiamos "ScratchJr" **blokais**. Komandų blokų rinkinys sudaro programą, dar vadinamą veikimo **scenarijumi**. Visų veikėjų scenarijai (programos) sudaro viso projekto (tam tikrai temai skirto darbo) programą.

Vaikai gali daryti garso įrašus, fotografuoti ir įtraukti garsus ar nuotraukas į savo kuriamas programas. Visa tai leidžia realiu laiku įrašyti įspūdžius, pavyzdžiui, mokomosios išvykos metu.

#### Reikalavimai sistemai

"ScratchJr" priemonė skirta "Android" (4.2 ir naujesnėms) ir "iOS" (8.0 ar naujesnėms) operacinių sistemų bent 7 colių įstrižainės planšetiniams kompiuteriams. Priemonė taip pat veikia "Chromebook" įrenginiuose. Ją galima parsisiųsti iš "Google Play", "Apple App Store", "Chrome Web Store" svetainių nemokamai (nuorodas galima rasti svetainėje *https://www.scratchjr. org/*).

Jei nėra galimybės dirbti planšetiniame kompiuteryje arba planšetinių kompiuterių yra nepakankamai, galima naudotis "ScratchJr" priemone staliniame arba nešiojamajame kompiuteryje (su "Windows" arba "Mac" grupės operacine sistema). Tam reikia parsisiųsti ir įdiegti "Android" sistemos emuliatorių (pvz., "MEmu", *http://www.memuplay.com*).

#### Veikėjai ir veiksmų laukas

Pradėjus naują "ScratchJr" projektą, pirmiausia reikia apmąstyti, kas vykdys komandas (veikėją ar veikėjus) ir kur vyks programuojami veiksmai (veiksmų laukas). "ScratchJr" aplinkoje yra didelis veikėjų ir veiksmų laukų piešinių pasirinkimas (paveiksle pateiktas siūlomų veikėjų rinkinio fragmentas), tačiau daugeliui vaikų kūrybiškai idėjų raiškai prireiks kitokių veikėjų ar veiksmų laukų – juos galima nupiešti arba pakoreguoti esamus veikėjus ar fonus.



Pradėjus naują projektą, tuščiame veiksmų lauke būna tik vienas numatytasis veikėjas – Katinas. Kitas veikėjas pridedamas paspaudus mygtuką su pliuso ženklu.



Jei norima pasirinkti veikėją tokį, koks jis siūlomas "ScratchJr" aplinkoje, spaudžiamas varnelės mygtukas, jei norima jį koreguoti, pažymimas veikėjas ir pasirenkamas teptuko mygtukas, jei norima atsisakyti veikėjo pasirinkimo, spaudžiamas kryžiukas. Jei norima pašalinti esamą veikėją iš veiksmų lauko (pvz., numatytąjį veikėją Katiną), reikia jį spustelėti ir palaikyti, tuomet paspausti atsiradusį raudoną kryžiuką.

Jei norima nupiešti veikėją nuo pradžių, pasirenkamas pradžioje siūlomas tuščias lapas ir spaudžiamas mygtukas su teptuku.



Pastebėsime, kad esamame veikėjų sąraše siūlomi veikėjų ruošiniai be veidų. Jie parengti tam, kad vaikas galėtų individualizuoti savo kuriamą projektą – nufotografuoti savo ar draugo veidą ir įdėti nuotrauką veikėjo tuščio veido vietoje. "ScratchJr" turima priemonė leidžia automatiškai įdėti nuotrauką į nurodytą piešinio sritį, pvz., veikėjo veidą (apie tai žr. toliau).



Pasirinkus redaguoti arba piešti naują veikėją, naudojama integruota "ScratchJr" piešimo priemonė.



Viršuje yra trys mygtukai: kairėje – atšaukti paskutinį veiksmą, grąžinti paskutinį atšauktą veiksmą, dešinėje – patvirtinti visus veiksmus ir įrašyti piešinį.

Kairėje vertikaliai išdėstyti piešimo objektų įrankiai (linija, apskritimas, stačiakampis, trikampis) ir jų linijų storių pasirinkimo mygtukai, apačioje – spalvų paletė, o dešinėje vertikaliai – veiksmų su piešiamais objektais įrankiai. Trumpai juos apibūdinsime iš eilės, iš viršaus žemyn.

	Objekto kontūro taisymo ir objekto tempimo priemonė. Pasirenkama priemo- nė, o tada objektas, kurio padėtį piešimo lauke arba kontūrą koreguojant taš- kus norima keisti.
Q	Objekto posūkio priemonė. Pasirenkama priemonė, o tada objektas, kurį nori- ma pasukti.
•	Kopijavimo priemonė. Pasirinkus kopijavimo priemonę ir pažymėjus objektą, sukuriama jo kopija.
×	Iškirpimo priemonė. Pažymėjus mygtuką su žirklėmis, o tada pažymėjus objektą, jis pašalinamas iš piešimo lauko.
0	Integruota fotografavimo priemonė. Pasirinkus fotografavimo priemonę ir spustelėjus bet kurią piešinio sritį, fotografuojama, o pasirinkta sritis užpildo- ma padaryta nuotrauka. Gali būti naudojama praturtinant fonus nuotraukomis ir kuriant individualizuotus veikėjus, pavyzdžiui, veikėjus, kurių veidas – vaiko veido nuotrauka.
	Spalvinimo priemonė. Pasirinkus priemonę ir spustelėjus piešinio sritį, ta sritis nuspalvinama paletėje pažymėta spalva.

Integruota fotografavimo priemonė leidžia individualizuoti veikėjus. Pažymėta piešinio sritis užpildoma nufotografuotu vaizdu. Atkreiptinas dėmesys, kad veikėjo ar veiksmų lauko piešinio negalima papildyti nuotrauka ar paveikslu iš failo. Yra galimybė įkelti tik realiu laiku, integruota fotografavimo priemone padarytą nuotrauką.

Veikėjų programuojami veiksmai atliekami veiksmų lauke.



Veiksmų lauką (foną) galima pasirinkti iš esamo rinkinio, nupiešti naują arba pakoreguoti esamą. Naudojama ta pati piešimo priemonė, kurią ką tik aptarėme. Fonui kurti ji paleidžiama pasirinkus fono piktogramą viršutinėje "ScratchJr" mygtukų juostoje, pažymėjus tuščią foną ir pasirinkus piešimo mygtuką su pavaizduotu teptuku.

# Praktinės užduotys

1. Pakoreguokite vieną iš esamų veikėjų: įdėkite nuotrauką, pakeiskite spalvas, pridėkite naujų detalių ir pan. Nupieškite šiam veikėjui skirtą veiksmų lauką, panaudodami kuo daugiau įvairių piešimo priemonių.

2. Kuo skiriasi veikėjas nuo objekto, nupiešto veiksmų lauko piešinyje?

# "ScratchJr" blokų apžvalga

"ScratchJr" programavimo kalbos blokai suskirstyti į šešias dalis ir nuspalvinti skirtingomis spalvomis, kad juos būtų lengviau atpažinti.

Blokų grupė	Aprašas
Veiksmų perjungimas ir aktyvinimas	Geltonos spalvos blokai skirti paleisti pa- rašytą programą ir valdyti veikėjų sąveiką.
Judesiai	Žydros spalvos blokai skirti veikėjo jude- siams (į kairę, į dešinę, į viršų, apsisuki- mas, grįžimas į pradinę vietą ir pan.) pro- gramuoti.
Išvaizda	Purpurinės spalvos blokai skirti veikėjų iš- vaizdai keisti (pasislėpti, pasirodyti, keisti dydį, rodyti teksto burbulą).
Garsai	Žalios spalvos blokai skirti garsui įrašyti, įrašytam ir numatytajam garsui groti.
Valdymas	Oranžinės spalvos blokai skirti veiks- mams kartoti, stabdyti, atidėti, keisti veiksmų atlikimo spartą.
Baigimas	Raudonos spalvos blokais nurodoma, kas vyksta programos atlikimo pabaigoje: baigiama, visada kartojama ar pereinama į naują puslapį– veiksmų lauką.

Nors blokai sugrupuoti į šias grupes, tačiau toks grupavimas yra sąlyginis. Kuriant tam tikrą projektą, paprastai naudojami įvairių grupių veiksmai. Su konkrečiais blokais ir jų naudojimu susipažinsime tolesniuose šios metodinės medžiagos skyreliuose.

# Praktinė užduotis

Parsisiųskite blokų piešinius spausdinimui iš "ScratchJr" kūrėjų svetainės: *http://scratchjr. org/pdfs/blocks.pdf*. Parenkite popierinių blokų ruošinius mokinių pirmiesiems žingsniams su "ScratchJr".

Pažintį su programavimu "ScratchJr" aplinkoje paprasčiausia pradėti valdant komandomis veikėjų judesius. Programuodami veikėjų veiksmus, vaikai įsisavina informatikos sąvokas algoritmas ir programa.

#### algoritmas

Baigtinė seka aiškiai suformuluotų veiksmų, kuriuos reikia atlikti kuriam nors uždaviniui išspręsti ar tikslui pasiekti.

#### programa

Kompiuteriu atliekamų veiksmų aprašas kuria nors programavimo kalba.

Paprastai tariant, programa – tai algoritmas, užrašytas programavimo kalba. "ScratchJr" atveju – piktogramine kalba, kurios programavimo konstrukcijos reiškiamos grafiniais paveikslėliais – blokais. Santykis tarp algoritmo ir programos vaikų įsisavinamas, pavyzdžiui, kai jie įvardija žodžiais, ką turės padaryti veikėjas ar veikėjai (algoritmas), o tada įgyvendina šiuos veiksmus programavimo aplinkoje (programa).

Kiekviena programa "ScratcJr" aplinkoje susieta su ją atliekančiu veikėju. Todėl prieš pradedant kurti programą, reikia pasirinkti veikėją, kuris atliks programą sudarančias komandas. Apačioje, žemiau blokų rinkinių atsiras veikėjas ir tuščias laukas, kuriame bus galima dėlioti veikėjo programą iš blokų (veikėjų komandų laukas).



Kuriant programą, reikia nurodyti, kada kiekvieno veikėjo veiksmų seka turės prasidėti. Veiksmų pradžios nurodymo blokai pasiekiami iš geltonos spalvos blokų rinkinio.



Dažniausi veiksmų pradžios variantai:

• spustelėjus veikėją (antras iš eilės geltonas blokas);

• paspaudus specialų programos pradėjimo ženklą – žalią vėliavėlę (pirmas iš kairės geltonas blokas).

Žalia vėliavėlė – specialus "ScratchJr" ženklas, esantis virš veiksmų lauko mygtukų juostoje, dešinėje. Jį paspaudus, pradedama programa, t. y. atliekami visi veikėjų veiksmai, kurie prasideda žalios vėliavėlės ženklu:



Judesių programavimo blokai yra žydros spalvos. Yra tokie judėjimo veiksmų variantai (iš kairės į dešinę atitinka žydros spalvos blokus):

- judėti nurodytą žingsnių skaičių į dešinę;
- judėti nurodytą žingsnių skaičių į kairę;
- judėti nurodytą žingsnių skaičių į viršų;
- judėti nurodytą žingsnių skaičių žemyn;
- pasukti veikėjo piešinį pagal laikrodžio rodyklę;
- pasukti veikėjo piešinį prieš laikrodžio rodyklę;
- pašokti nurodytą žingsnių skaičių;
- grąžinti veikėją į jo pradinę poziciją.



Jei norime, kad veikėjas pradėtų judėti, kai paspausime žalios vėliavėlės ženklą, jo judesių programos (scenarijaus) pavyzdys galėtų atrodyti taip:



Ši programa "sako", kad Varlė judėtų tris žingsnius į kairę, pašoktų 2 žingsniais, judėtų tris žingsnius į dešinę, vėl pašoktų ir visiškai apsisuktų pagal laikrodžio rodyklę. Raudonas blokas pabaigoje žymi veikėjo veiksmų pabaigą (tai nebūtinas blokas).

Judėjimo, šoktelėjimo žingsniai nurodomi specialiais matavimo vienetais, kuriuos galima pamatyti įjungus veiksmų lauko tinklelį (tam reikia spustelėti tinklelio piktogramą virš veiksmų lauko – antra iš eilės piktograma).

Įjungę tinklelį, matome, kad kiekvienas veikėjas veiksmų lauke turi koordinates. Laukas padalytas į 20 × 15 langelių. Veikėjo aktyvus taškas yra veikėjo piešinio viduje. Kai veikėjas pažymėtas, rodomos šio veikėjo aktyvus taškas ir koordinatės. Pavyzdžiui, toliau pateiktame pavyzdyje Katino koordinatės yra (3; 6), aktyvus jo taškas pažymėtas melsva spalva.



Nurodžius, kiek žingsnių judės veikėjas, žingsniai pradedami skaičiuoti būtent nuo šio aktyvaus taško. Vienas žingsnis lygus vienam tinklelio langeliui.

Veikėjo posūkio matavimo vienetai – 1/12 apskritimo. Norint, kad veikėjas atliktų visą posūkį, nurodoma 12. Norint, kad veikėjas "atsistotų ant galvos", nurodoma 6.

Judesius galima valdyti nurodant neigiamą žingsnių skaičių. Tokiu atveju veikėjas judės priešinga kryptimi negu rodo bloke pavaizduota rodykle.

Jei norima, kad veikėjas pasisuktų veidu kita kryptimi, pasirenkamas kryptį nurodantis blokas, o žingsnių skaičius – nulis.

Jei veikėjui nurodyta judėti žingsnių skaičių tokį, kad jis išeitų už veiksmų lauko ribų, jis tęsia savo judesį iš priešingos veiksmų lauko pusės. Tuo patogu naudotis, pavyzdžiui, kuriant akvariumo animaciją: žuvytė, pasiekusi dešinį akvariumo kraštą, atsiranda vėl iš kairės ir plaukia toliau.

Jei norima, kad veikėjas niekada neišeitų už veiksmų lauko ribų, reikia susikurti veikėjus, kurie "apribotų" veiksmų lauką, ir programuoti kelių veikėjų sąveiką (žr. skyrelį "Veikėjų sąveika. Susidūrimo įvykis").

Veikėjų judėjimo greitį galima valdyti naudojant oranžinės spalvos blokus, pavyzdžiui:



**Patarimas.** Su komandų vykdymu ir veikėjų judesiais vaikus galima pradėti supažindinti be kompiuterio, o tik tada pereiti prie programavimo skaitmeninėje priemonėje. Programas galima dėlioti iš popierinių blokų (organizavus grupinius arba individualius mokinių darbus).

# Praktinės užduotys

1. Išnagrinėkite ir išbandykite veiklą "Ar visi juda vienodu greičiu?", parsisiųskite popierinius blokus, atlikite veiksmus, aprašytus užduočių lape. Išbandykite veiklą su mokiniais.

2. Išnagrinėkite veiklą "Atspėk komandas". Sugalvokite keletą trumpų užduočių šiai veiklai papildyti.

3. Išnagrinėkite veiklą "Ištaisyk programą". Sukurkite daugiau trumpų programų fragmentų, kuriuose vaikai turės rasti klaidų.

# Garsai

Žalių blokų rinkinyje pateikiamos garsų įrašymo ir jo grojimo komandų priemonės, taip pat numatytojo garso "Pop" grojimo komanda. Tai nuostabi galimybė vaikams išreikšti save, įrašyti savo įspūdžius, pasakojimus, imituoti įvairių veikėjų balsus ir pan.

Garso įrašymo priemonė integruota į garsų komandų blokų rinkinį (žalios spalvos blokai).



Pasirinkus veikėją ir spustelėjus žalią mikrofono bloką su punktyriniu kontūru, atsiveria langas – jame paspaudus mygtuką su raudonu skrituliuku, pradedamas balso įrašymas. Įrašymas baigiamas spustelėjus varnelę (patvirtinimo mygtuką).



Įrašius garsą, žalių komandų blokuose atsiranda blokas su ką tik įrašyto garso įrašo grojimo komanda (žalias blokas su mikrofono vaizdu, be punktyrinio kontūro).

Pavyzdžiui, norint, kad veikėjas pradėtų pasakoti (grotų įrašytą garsą) paspaudus jį, naudojama geltonų (paleidimo) blokų komanda "paspaudus" ir žalias įrašyto garso grojimo komandos blokas.



O norint, kad veikėjas grotų įrašytą garsą paleidus programą (t. y. paspaudus žalią vėliavėlę), naudojama tokia blokų seka.



Žalios spalvos blokų rinkinyje esantis blokas "pop" leidžia įdėti numatytojo trumpo garso-signalo grojimą.

# Praktinė užduotis

Atlikite veiklą "Pavasario žiedai". Išbandykite garso įrašymo priemonę ir į programą įtraukite sukurtus trumpus garsinius pasakojimus apie pavasario gėles.



Jei norime, kad veikėjas atliktų veiksmus nuosekliai, dėliojame komandas į vieną liniją – kiekviena dešiniau pridėto bloko komanda atliekama po kairėje esančio bloko komandos. Tai nuoseklusis komandų vykdymas.



Kas atsitiks, jei dvi skirtingas komandų sekas pridėsime po to paties veiksmų pradžios bloko, pavyzdžiui, po žalios vėliavėlės? Šiuo atveju susiduriame su informatikos sąvoka lygiagretusis komandų vykdymas. Nurodytos komandų sekos vykdomos abi vienu metu, lygiagrečiai. Pavyzdžiui, jei turime tokias dvi komandas sekas (žr. toliau pateiktą pavyzdį), pradedamas paspaudus vėliavėlės ženklą, tai veikėjas atliks šokį (judesius, aprašytus žydrais blokais) su "akompanimentu" (žaliais blokais grojamas "Pop" garsas. Raudonas blokas pabaigoje rodo, kad komandų seka kartojama nuolat (žr. toliau apie begalinį ciklą).



Kitas pavyzdys. Palietus veikėją Viščiuką, vykdomos dvi veiksmų sekos: grojamas garso įrašas, pavyzdžiui, veikėjo balsu sakomi žodžiai ir vienu metu atliekami keli judesiai.



Tinkamai pritaikius lygiagretųjį vykdymą, kuriant konkretų projektą, galima sukurti sudėtingesnius veikėjų veiksmus.

# Praktinė užduotis

Išbandykite veiklą "Veikėjų šokis". Sugalvokite šios veiklos modifikaciją, papildydami ją papildomomis lygiagrečiojo vykdymo komandomis kitiems šokio efektams išgauti.

# Veiksmų kartojimas

Ankstesniame skyrelyje jau susidūrėme su veiksmų kartojimu. Jei tam tikrą veiksmų seką reikia pakartoti daugiau kaip vieną kartą, siūloma naudoti ciklo konstrukciją.

#### ciklas

Programavimo kalbos konstrukcija kartojamiems veiksmams užrašyti.

Ciklai "ScratchJr" aplinkoje yra dviejų tipų: žinomo kartojimų skaičiaus ciklas ir begalinis ciklas. Atitinkami blokai pateikiami šalia šių sąvokų apibrėžčių.



žinomo kartojimų skaičiaus ciklas

Programavimo konstrukcija veiksmų sekai kartoti žinomą kartų skaičių.



**begalinis ciklas** Niekada nesibaigiantis ciklas.

Pavyzdžiui, jei norėtume, kad Varlė kartotų anksčiau aprašytą judesių grupę tris kartus, jos programa turėtų būti tokia:



Žinomo kartojimo skaičiaus ciklo oranžinis blokas gaubia judesių komandų seką. Skaičius, nurodytas oranžiniame ciklo bloke, yra 3. Tai reiškia, kad ciklu "apgaubta" komandų seka kartojama tris kartus.

Mokiniai galėtų palyginti šių dviejų programų rezultatus ir įsitikinti, kad rezultatas yra tas pats:



Jei veiksmus reikia kartoti nelabai daug kartų, galima išsisukti ir be ciklo, tačiau jei kartoti tektų, pavyzdžiui, 10, 20 ar daugiau kartų, būtų akivaizdu, kad naudingiau taikyti ciklą. Jei norime kartoti veiksmus visą laiką, kol vykdoma programa, naudojame begalinį ciklą.

Begalinis, niekada nesibaigiantis ciklas nereiškia, kad jo negalima sustabdyti. Jis vykdomas tiek, kiek vykdoma programa, vadinasi, ir baigiamas tada, kai baigiama programa. Tačiau šiuo atveju būtina specialiai imtis veiksmų programai užbaigti – sustabdyti. Tai galima atlikti paspaudus raudoną mygtuką, kuris matomas žalios vėliavėlės vietoje vykdant projektą, virš veiksmų lauko, dešinėje.



#### Praktinė užduotis

Pateikite keletą pavyzdžių, kai komandas racionaliau vykdyti pritaikius ciklo konstrukciją negu be jos. Išbandykite šiuos pavyzdžius "ScratchJr" aplinkoje.

#### Sąlygos ir įvykiai

Programavime ir veiksmų automatizavime, kaip ir tikrame gyvenime, nuolat tikrinamos tam tikros sąlygos. Jei sąlyga tenkinama, vykdomi vieni veiksmai, jeigu ji netenkinama, vykdomi kiti.

Programavime sąlygos reiškiamos sąlyginiais sakiniais.

#### sąlyginis sakinys

Programavimo kalbos valdymo konstrukcija, apibrėžianti sąlygą ir veiksmus, kurie turi būti atliekami atsižvelgiant į tai, ar sąlyga tenkinama, ar ne.

"ScratchJr" aplinkoje sąlyginių sakinių nerasime, kadangi aplinka pritaikyta jaunesniojo amžiaus mokiniams, o norint užrašyti daugumą sąlygų, reikia žinoti kintamojo ir jo reikšmės sąvokas (o tai būtų pernelyg sudėtinga pradinukams). Todėl čia sąlyginius sakinius keičia įvykių programavimas.

# Įvykis

Veiksmas, į kurį programa turi reaguoti, pavyzdžiui:

- klavišo paspaudimas,
- pelės patraukimas,
- duomenų gavimas jutikliu,
- kitų programų pranešimai,
- programos signalas apie klaidą.

Iš esmės įvykius galime traktuoti kaip sąlygas. Pateiktos įvykio sąvokos apibrėžtyje minimus įvykių pavyzdžius galima suprasti kaip sąlygas: "jei paspaustas klavišas, tai…", "jei patraukta pelė, tai…", "jei jutiklis gavo duomenų, tai…", "jei kita programa pateikė pranešimą, tai…", "jei programa pateikė signalą apie klaidą, tai…".

"ScratchJr" aplinkoje vykdant programą nuolat stebima, ar neįvyko kuris nors galimas įvykis. Jei įvyko, vykdoma nurodyta komandų seka. Pavyzdžiui, palietus veikėją (įvykus veikėjo paspaudimo įvykiui – geltonas blokas), vykdoma komandų seka: vidutiniu greičiu (oranžinis blokas) judėti į dešinę, žemyn, į dešinę ir pašokti (keturi žydros spalvos judesių blokai).



Pagrindinius "ScratchJr" įvykius galima nusakyti naudojant geltonus blokus:

- paspaudus žalios vėliavėlės ženkliuką;
- palietus (paspaudus) veikėją;
- susidūrus su kitu veikėju;
- gavus pranešimą iš kito veikėjo.

# Veikėjų sąveika

Daugumoje vaikų darbelių, kuriamų "ScratchJr", dalyvaus ne vienas, o keli veikėjai. Kai veikėjų yra daugiau kaip vienas, reikia pagalvoti apie jų sąveiką ir veiksmų suderinimą. Sąveikai įgyvendinti pirmiausia reikalingi geltoni įvykių blokai ir oranžiniai valdymo blokai.



Veikėjų sąveikai kurti naudojami įvykiai. Pagrindiniai įvykiai sąveikai yra: susidūrimas, laukimas, pranešimų siuntimas. Juos aptarsime tolesniuose skyreliuose.

# Susidūrimo įvykis

Daugeliui darbelių gali prireikti veikėjų "susidūrimo" programavimo. Pavyzdžiui, skrendantis drugelis, palietęs dramblį, keičia skrydžio kryptį.

Pastebėsime, kad veikėjai gali būti nebūtinai gyvi objektai (net numatytajame "ScratchJr" veikėjų rinkinyje siūlomi veikėjai – baldai, namai, automobiliai ir pan.). Pavyzdžiui, jei norėtume imituoti veikėjo įėjimo į pilį veiksmą, mums prireiktų dviejų veikėjų (pavyzdžiui, pilies ir Katino) ir toliau pateikiamų komandų (žr. pav.).



Katino veiksmai pradedami vykdyti paspaudus žalios vėliavėlės ženkliuką. Naudodami veiksmų lauko tinklelį, nustatome, kad Katinas paėjęs keturis žingsnius link pilies (judėjimas į viršų), susidurs su ja, ir bus vykdoma susidūrimo įvykio (geltonas blokas) komandų seka. Įvykus susidūrimo įvykiui, šiuo atveju pasirinkome komandas iš purpurinių (išvaizdos) blokų rinkinio: Katinas keičia dydį (mažėja, t. y. imituojamas Katino ėjimas į tolį) ir dingsta (imituojamas Katino įėjimas į pilį).

Aukščiau minėto Drugelio ir Dramblio pavyzdys galėtų atrodyti taip:



Pradėjus programą (paspaudus žalią vėliavėlę) Drugelis skrenda 10 žingsnių į kairę. Palietęs Dramblį (geltonas susidūrimo įvykio blokas), jis greitai (oranžinis greičio keitimo blokas) juda 12 žingsnių priešinga kryptimi (į dešinę). Dramblys taip pat gali vykdyti įvairias komandas, pvz., atlikti savo judesius.

# Laukimas

Vienas paprastesnių, bet primityvesnių būdų, kaip įgyvendinti veikėjų sąveiką, – naudoti laukimo komandą. Pavyzdžiui, vaikams pirmą kartą kuriant veikėjų dialogą galima pasiūlyti pasinaudoti laukimo komanda, kad vieno veikėjo "kalbėjimo" metu kitas veikėjas lauktų nurodytą laiko intervalą, nepradėtų "kalbėti" savo dialogo dalies. Toks veikėjų sąveikos programavimas labiau tiktų jaunesniojo amžiaus vaikams. Laukimo komanda pasiekiama iš oranžinių (valdymo) blokų srities. Skaičius joje nurodo, kiek sekundės dešimtųjų dalių reikia laukti. Pavyzdžiui, 10 reiškia 1 sekundės laukimą.



Toliau pateikiamas veikėjų dialogo programos fragmentas, kai sąveika įgyvendinama naudojant laukimo komandą.



Paspaudus žalios vėliavėlės mygtuką (pradėjus programą), berniukas "sako" tekstą, rodomą teksto burbule (purpurinis blokas). Mergaitė, pradėjus vykdyti programą (žalios vėliavėlės blokas), laukia 2 sekundes (skaičius 20 oranžiniame laukimo bloke) ir tik tada pradeda sakyti savo tekstą (purpurinis teksto burbulo blokas).

Laukimo komanda naudojama ir daugeliu kitų atvejų, pavyzdžiui, norint natūraliau imituoti to paties veikėjo veiksmus, kad veiksmai neatrodytų "susilieję", mechaniški. Pavyzdžiui, programuojant veikėjo judesius, ne iš karto daryti posūkį po žingsnio, o po tam tikros pauzės.

# Pranešimų siuntimas

Aptarta laukimo komanda yra ribotas sąveikos programavimo būdas, nes ne visada galima tiksliai numatyti, kiek laiko bus vykdomi veikėjo veiksmai. Racionalesnis būdas – naudoti "ScratchJr" pranešimų siuntimo tarp veikėjų įvykius. Pranešimų siuntimo ir gavimo komandos pateikiamos geltonų blokų rinkinyje.



Programuojant veikėjų sąveiką, vaikams svarbu pabrėžti, kad kiekvienas atskiras veikėjas vykdo savo komandas, nekreipdamas dėmesio į kitų veikėjų veiksmus. Tačiau, jei reikia suderinti kelių veikėjų veiksmus (pavyzdžiui, sukurti dialogą), jie turi tarpusavyje komunikuoti. Panašiai kaip ir gyvenime, kai dirbame bendrą darbą kartu. Tik tarp programinių veikėjų toks komunikavimas vyksta komandomis – siunčiant pranešimus. Pranešimo gavimo įvykis gali būti sąlyga, kad veikėjas pradeda atlikti tam tikrą komandų seką. Tam, kad galima būtų kurti keletą sąveikos iteracijų (tarkime, dialogą tarp veikėjų), "ScratchJr" yra skirtingos pranešimų siuntimą ir gavimą vaizduojančių vokų spalvos (šešios spalvos).



Veikėjų sąveikos programavimą pademonstruosime dialogo kūrimo pavyzdžiu. Tarkime, kad turime du veikėjus: Tomą ir Simoną. Pradėjus programą, Tomas pasisveikina. Tam, kad Simona atsakytų į Tomo pasisveikinimą, Tomas po pasisveikinimo žodžių turi jai išsiųsti pranešimą.

Pavyzdyje Tomas pradeda dialogą paleidus programą – paspaudus žalią vėliavėlę. Jo pokalbio tekstas vaizduojamas teksto burbuluose, kuriuos galima rasti purpurinių komandų blokų rinkinyje. Tam, kad Simona sureaguotų, pradėtų "kalbėti" iš karto po Tomo teksto, Tomas išsiunčia jai pranešimą (pavyzdyje – oranžinės spalvos vokas). Sudėliota trijų minėtų blokų seka pavaizduota paveiksle.



Veikėjai, esantys viename veiksmų lauke, gali siųsti vienas kitam ne daugiau kaip šešis pranešimus – "ScratchJr" priemonėje yra šešios laiškų vokų spalvos. Veikėjų siunčiami pranešimai skiriami spalvomis, kad veikėjas galėtų atskiri vieną pranešimą nuo kito, ir gavęs kitą pranešimą galėtų atlikti kitus veiksmus.

Keleto dialogo iteracijų pavyzdys įgyvendinamas skirtingomis spalvomis vaizduojamų pranešimų siuntimu.



Tomas pradeda pokalbį, o tęsia tada, kai gauna žalią arba geltoną pranešimo voką. Simona pradeda pokalbį tada, kai gauna oranžinį voką, o pasisveikinusi išsiunčia geltoną voką (vadinasi, į geltoną voką Tomas reaguos sakydamas savo antrąją dialogo dalį). Gavusi iš Tomo raudoną voką, Simona sako dialogo tęsinį ir siunčia Tomui žalią voką. Trečioje vietoje Tomas reaguoja į žalią voką ir pasakęs savo žodžius siunčia mėlyną voką. Reaguodama į mėlyną

voką, Simona sako baigiamąją dialogo dalį.

Anksčiau pavaizduotame pavyzdyje matome, kad Simonai atsisveikinus su Tomu programa baigiama – įdedamas raudonas programos pabaigos blokas.

Dialogą galima papildyti laukimo komanda po kiekvieno sakomo teksto.



# Praktinės užduotys

1. Suprogramuokite veikėjų dialogą dviem būdais: naudodamiesi laukimo komanda ir naudodamiesi pranešimų siuntimo komandomis. Palyginkite rezultatus.

2. Praktiškai išbandykite veiklas "Aritmetiniai veiksmai" ir "Pasaulio kryptys". Identifikuokite laukimo, susidūrimo, pranešimų siuntimo įvykius. Nusakykite alternatyvius šių veiklų scenarijų įgyvendinimo būdus.

# Puslapiai. Veiksmų lauko keitimas

Kuriant sudėtingesnius projektus, dažnai nepakanka vieno veiksmų lauko, todėl animaciją galima skaidyti į keletą dalių su skirtingais veiksmų laukais. Tam "ScratchJr" aplinkoje naudojami puslapiai. Puslapis iš esmės ir yra naujas veiksmų laukas. Viename projekte iš viso gali būti ne daugiau kaip keturi puslapiai. Manoma, kad tokio puslapių skaičiaus turėtų užtekti pradinių klasių mokinių darbams. Projektas iš kelių puslapių primena knygą.

Kurdami kelių puslapių projektus, mokiniai nejučia pradeda pažintį su informatikos sąvoka galiojimo sritis.

#### galiojimo sritis

Sritis, ribojanti aprašų, formatų, žymėjimų, nuostatų ir kitokių susitarimų galiojimą. Programavime – autonominė programos aprašų dalis.

Visos vieno puslapio komandos galioja tik tame puslapyje (puslapis – tai tarsi programa programoje). Pavyzdžiui, jei veikėjai komunikuoja siųsdami pranešimus ("ScratchJr" programoje vaizduojami šešių spalvų vokais), tai geltonas vokas viename puslapyje gali reikšti vieną pranešimą, o kitame – kitą (t. y. puslapiai ir komandos juose yra nepriklausomi, pranešimo vokų spalvos juose gali kartotis, bet reikš skirtinus dalykus). Taip pat svarbu nurodyti, kurio veikėjo veiksmų pabaigoje įvyks perėjimas į kitą puslapį, kurio veikėjo veiksmų pabaigoje bus baigiama paskutinio puslapio programa.

#### Naujo puslapio kūrimas

Naujas puslapis kuriamas paspaudus mygtuką su pliuso ženklu lango dešinėje po puslapį vaizduojama miniatiūra. Toliau pateiktame paveiksle sukurti trys puslapiai (pliuso mygtukas visuomet būna žemiau paskutinio sukurto puslapio miniatiūros).



Pažymėjus atitinkamą puslapį dešinėje, atsiveria jo veiksmų laukas ir veikėjai. Veikėjų veiksmai galioja tik viename puslapyje. Pasakojimas tęsiamas pereinant iš vieno puslapio į kitą.

# Perėjimas tarp puslapių

Perėjimas iš vieno puslapio į kitą paprastai daromas įvykus tam tikram įvykiui, pvz.:

- gautas pranešimas (tam tikros spalvos vokas);
- paliestas veikėjas;
- du veikėjai susidūrė.

Perėjimo į kitą puslapį komandas galima rasti baigimo (raudonų) blokų rinkinyje, bet tik tada, kai projekte yra įterptas bent vienas papildomas puslapis.



Pavyzdžiui, toliau pateikta dviejų blokų seka reiškia, kad į nurodytą kitą (trečią) puslapį pereiti reikia tada, kai įvyks "susidūrimo" su obuoliu įvykis, tarkime, kai Katinas "paims" obuolį.



Toliau pateikiamas dialogo fragmento komandų pavyzdys su perėjimu į kitą puslapį.



Paleidus puslapio programą, berniukas sako teksto burbule rodomus žodžius ir siunčia pranešimą, kai jo dialogo dalis baigta. Raudonas vokas vaizduoja pranešimą, gautą iš kito veikėjo (pvz., berniuko pašnekovo). Tuomet berniukas laukia vieną sekundę, ir vyksta perėjimas į kitą puslapį, kuriame keičiamas veiksmų laukas.

Kai projekte yra keletas puslapių, susietų tarpusavyje raudonais blokais "pereiti į puslapį", visi scenarijai gali būti leidžiami visiškai automatiškai, pradėjus programą pirmame puslapyje, pavyzdžiui, paspaudus žalios vėliavėlės ženklą. Įvykus automatiniam perėjimui į kitą puslapį, automatiškai pradedamos vykdyti komandų sekos, pradedamos žalios vėliavėlės ženklu. Jei norima rankiniu būdu įvykdyti atskiro puslapio komandas, reikia tą puslapį pažymėti (dešinėje) ir paspausti žalią vėliavėlę. Norint projektą paleisti nuo pradžių, reikia pažymėti pirmą jo puslapį. Pridėtų projekto puslapių eiliškumą galima keisti tiesiog tempiant į reikiamą eilės vietą dešinėje esančias puslapių piktogramas.



Pakeitus puslapių eiliškumą, "ScratchJr" automatiškai pakeis puslapių numeraciją perėjimo į kitą puslapį komandų blokuose (jei tokių buvo). Taigi baigus esamą komandą bus pereita prie esamoje programoje numatyto puslapio numerio.

# Veikėjų ir jų scenarijų kopijavimas

Kuriant keleto puslapių projektą, puslapiuose paprastai dalyvauja tam tikra grupė tų pačių veikėjų. Dažnai ir dalis jų veiksmų kartojama kitame puslapyje. Todėl būtų patogu kopijuoti veikėją su visais jo veiksmais iš vieno puslapio į kitą. Tam reikia, pasirinkus veikėją iš veikėjų sąrašo kairėje, nutempti jį į atitinkamą puslapį puslapių sąraše dešinėje, kaip parodyta paveiksle. Tas pats veiksmas praverstų, jei norima tame pačiame puslapyje padaryti daugiau to paties veikėjo su jo veiksmais kopijų, pavyzdžiui, miške sudėlioti keletą grybų.



Visi kopijuojamo veikėjo veiksmai ir įrašyti garsai nukopijuojami kartu su juo. Tačiau nukopijuoto veikėjo komandos ir originalaus veikėjo komandos tampa nepriklausomos: pakeitus veikėjo komandas viename puslapyje, jos automatiškai nesikečia kitame puslapyje. Keitimus reikia daryti atskirai.

Norint nukopijuoti veikėjo veiksmus toje pačioje programavimo srityje, t. y. pakartoti tam pačiam veikėjui, blokų seka tempiama į atitinkamo veikėjo piktogramą veikėjų sąraše (kairėje). Taip pat galima nukopijuoti veiksmų seką ir kitam veikėjui, esančiam tame pačiame puslapyje. Vienintelė išimtis – vieno veikėjo įrašyti garsai nekopijuojami prie kito veikėjo. Jei veikėjas turėjo įrašytų garsų, nukopijavus jie pakeičiami numatytaisiais "Pop" garsais.

# Praktinės užduotys

1. Praktiškai išbandykite veiklos "Surask paslėptus objektus" projektą. Sukurkite šio žaidimo antrą lygį, kuriame būtų naudojamas kitas veiksmų laukas, o objektus rasti būtų sunkiau. Perėjimas į antrą lygį gali būti įgyvendinamas paspaudus tam tikrą veikėją-mygtuką.

2. Sukurkite pasakojimą iš bent trijų puslapių (pasakojimo pradžia, pagrindinė dalis ir pabaiga), kuriame perėjimas tarp visų puslapių įgyvendinamas automatiškai (baigus atlikti tam tikro veikėjo veiksmus).