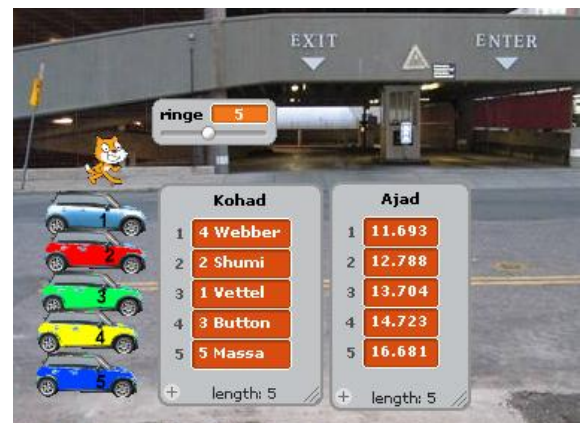


Loendid Scratchis



Sisu

Loendite käsitlemine Scratchis.....	3
Loendi olemus ja selle loomine	3
Loendi ja selle elementide omadused ning meetodid	4
Käsuplokid tegevuste määramiseks loendite elementidega ja loenditega	4
Põhitegevused ja tüüpalgoritmid loendite ja nende elementidega.....	4
Loendi elementide järjestikune töötlemine	4
Näide „Keele test“. Testid ja viktoriinid	5
Näide „Tõlkimine“	6
Dialogid, vestlused ja koomiksid	6
Loendite elementide loomine programmiga	7
Loendi elementide sisestamine klaviatuurilt	7
Loendi loomine juhuarvudest	7
Mõned tüüpalgoritmid loenditega	8
Loendi elementide summa ja aritmeetiline keskmine	8
Maksimaalne element loendis ja selle järjenumber	8
Loendite kasutamine tööks objektidega	Error! Bookmark not defined.
Näide. Autode võidusõit.....	8

Loendite käsitlemine Scratchis

Loendi olemus ja selle loomine

Loend (*List*) on järjestatud mäluväljade (muutujate) kogum, mis tähistatakse ühe nimega. Loendi elementidele viidatakse nime ja indeksi (järjenumbri) abil. Loendi üheks analoogiks võiks olla näiteks tabeli tulp (veerg). Näiteks võib allpool toodud statistiliste andmete tabeli iga tulp vaadelda loendina. Loendiks võiks olla tulp **Pikkus**, selle elementidele võib formaalselt viidata järgmiselt: **Pikkus₁**, **Pikkus₂**, **Pikkus₃** jne.

Meeskond: **Torm**

Jrn	Eesnimi	Perenimi	Sünniaasta	Pikkus	Kaal	...
1	Rein	Aavik	1990	192	76	
2	Andres	Heinsoo	1991	186	69	
3	Rasmus	Kasemets	1990	190	77	
4	Kristjan	Kuuspakk	1989	179	65	
5	Markus	Merisoo	1992	181	70	
6	Paul	Puusepp	1991	194	81	
...	

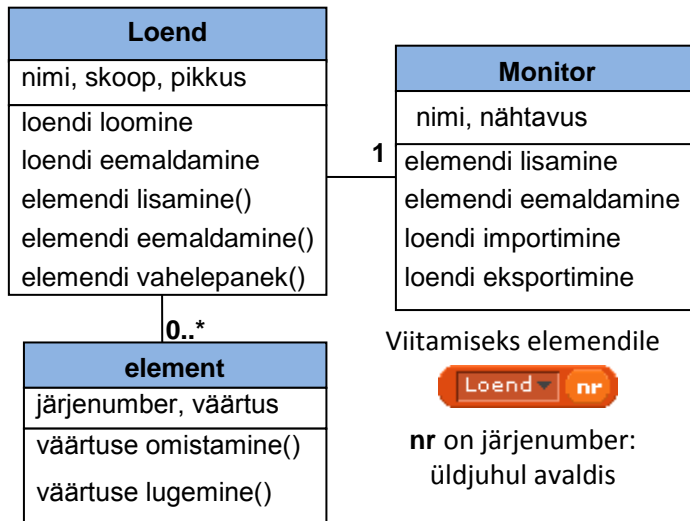
Käsuga **Tee loend** luuakse Scratchis tühi loend. Kasutajalt küsitakse **loendi nimi** ja **skoop**. Loendi nime valimisel võib lähtuda samadest kriteeriumidest nagu muutujate puhul. Nii nagu muutujad, võivad ka loendid olla globaalsed või lokaalsed. Laval tekib loendi tühi monitor ning süsteemis fikseeritakse loendi



nimi ja skoop. Koos esimese loendiga ilmuvad ka käsuplokid tegevuste täitmiseks loendite ja loendite elementidega: **lisamine**, **eemaldamine**, **sisestamine** ja **asendamine**.

Loendi elemente saab lisada, muuta ja eemalda ka nõ „käsitsi“ ning sageli nii ka toimitakse. Alustades tühjast monitorist lisatakse järjest elemente, vajadusel ka parandades (muutes) ja eemaldades neid. Lisamiseks klõpsata „+“ monitori alumises servas. Ilmub tühi väli, kuhu saab sisestada väärtuse ning see salvestatakse automaatselt arvuti mälu antud loendile eraldatud piirkonnas. Elementide numeratsioon algab alati ühest ja suureneb sammuga üks. Olemasoleva elemendi muutmiseks aktiveeritakse see hiireklõpsuga ning muudetakse väärtust või sisestatakse uus. Kui klõpsata aktiivse elemendi väljas olevat ristikest, eemaldatakse element. Loendi elemendid saab **importida** tekstifailist ja **eksportida** tekstifaili. Vastavad käsud ilmuvad, kui teha paremklõps loendi monitoril.

Loendi ja selle elementide omadused ning meetodid



Loendi nimest ja **skoobist** oli juttu eespool. **Loendi pikkus** on elementide arv loendis. Tegemist võib olla muutuva suurusega, sest elemente saab lisada ja eemaldada. Loend võib olla ka tühi, siis on pikkus 0.








Loendite loomine ja eemaldamine saab toimuda ainult projekti loomise faasis. Programmi täitmise ajal neid tegevusi täita ei saa.

Programselt saab lisada elemente loendi lõppu, panna vahele ning eemaldada.

Vastavad käsuplokid on toodud allpool olevas tabelis

Loendi element vastab oma olemuselt muutujale – tegemist on mäluväljaga. Oluliseks erinevuseks on, et loendi elementidel ei ole nimesid, neile vastab järjenumber, mis määrab elemendi asukoha loendis. Elemendi väärtuseks võib olla arv, tekst, tõeväärtus või objekt (sprait, helikliip, kostüüm) nimi, st loendite elemente saab (nagu muutujaidki) kasutada ka viitamiseks objektidele.

Käsuplokid tegevuste määramiseks loendite elementidega ja loenditega

Käsuplokk	Selgitus
	Lisab elemendi (väärtuse) loendi lõppu. Loend esitatakse nime abil. Lisatav element võib olla esitatud avaldisena.
	Eemaldab ühe või kõik väärtused loendist. Loend esitatakse nime abil. Elemendi määrab järjenumber; võib ka valida: 1, viimane, kõik .
	Sisestab (paneab vahele) elemendi antud kohale loendis. Võib anda järjenumbri või valida menüüst: 1, viimane, mõni .
	Asendab antud numbriga elemendi loendis avaldise väärtusega. Võib anda järjenumbri või valida menüüst: 1, viimane, mõni .
	Viitab antud numbriga elemendile loendis. Võib anda järjenumbri või valida menüüst: 1, viimane, mõni .
	Elementide arv loendis.
	Tagastab väärtuse tõene , kui loend sisaldab elementi, mis on võrdne avaldise väärtusega

Väärtuse omistamine loendi olemasolevale elemendile (st. väärtuste muutmine programmi täitmisel) toimub ploki [asenda ...], mis vastab omistamiskäsule [**võta muutuja = avaldis**]

Põhitegevused ja tüüpalgoritmid loendite ja nende elementidega

Loendi elementide järjestikune töötlemine

Loenditele on iseloomulik, et nende elementidega täidetakse ühetüübilisi tegevusi ja elemente töödeldakse (kasutatakse) järjest. Selleks saab kasutada määratud korduste arvuga kordust, muutes selles elemendi

järjenumbrit (indeksit) alates näiteks ühest kuni loendi pikkuseni sammuga üks. Põhimõtteliselt võib kasutada erinevaid vahemikke ja ka ühest erinevat sammu.

Näide. „Keele test“. Testid ja viktoriinid

Programm kuvab järjest eestikeelseid sõnu. Kasutaja peab sisestama nende vasteid inglise keeles. Programm kontrollib vastuseid, teeb kindlaks õigete vastuste arvu ja protsendi.

Kahes loendis on sõnad eesti ja inglise keeles (sama tähendusega sõnad on sama järjenumbriaga). Loendid on juba loodud näiteks „käsitsi“, nende nimedeks on **eesti** ja **inglise**. Elementide (sõnade) arv loendites võib muutuda; loendite pikkused on muidugi võrdsed.

eesti	inglise
1 kass	1 cat
2 koer	2 dog
3 hiir	3 mouse
4 lehm	4 cow
5 hobune	5 hourse
6 karu	6 bear
7 rebane	7 fox

Andmed

Loendid:

eesti (sõnu), inglise (sõnu)

Muutujad:

sõnu – sõnade arv

õigeid – õigete vastuste arv

vastus – kasutaja vastus

nr – elemendi järjenumber

protsent – õigete protsent

Keele test. Algoritm

sõnu = eesti.pikkus

õigeid = 0

nr = 1

kordus sõnu korda

 kui eesti(nr)

 loe vastus

 kui vastus = inglise(nr) siis

 õigeid = õigeid + 1

 muidu

 kui „Vale!“

 lõpp kui

 nr = nr + 1

lõpp kordus

protsent = õigeid / sõnu * 100



Programm teeb kindlaks elementide (sõnade) arvu loendites, kasutades loendi **eesti** omadust **pikkus**, ning omistab selle muutujale **sõnu**. Seejärel omistatakse algväärtused muutujatele **õigeid** ja **nr**. Korduses kuvatakse järjest sõnu loendist **eesti**, kasutades viitamiseks elementidele indeksit (muutujat) **nr**. Kasutaja vastust võrreldakse sama numbriga sõnaga loendis **inglise**. Kui vastus on õige, suurendatakse muutuja **õigeid** väärtust, vastupidisel juhul kuvatakse teade „Vale!“. Korduse lõpus suurendatakse muutuja **nr** väärtust ühe võrra, et järgmisel kordamisel viidata järgmisele elemendile. Kui kõik sõnad on läbitud, leitakse õigete vastuste protsent.

NB! Algoritmides kasutatakse viitamiseks loendi elementidele konstruktsiooni **nimi(indeks)**, kus **nimi** on loendi nimi ja **indeks** on elemendi järjenumber: **eesti(nr)**, **inglise(nr)**.

Analoogset lähenemisviisi võib kasutada suvaliste testide ja viktoriinide koostamiseks. Ühes loendis on küsimused, teises õiged vastused. Programm kuvab järjest küsimusi esimesest loendist, loeb kasutaja vastuseid ja võrdleb teises loendis olevate vastustega.

Näide „Tõlkimine“

Programm tõlgib sõnu inglise keelest eesti keelde. Programm loeb inglisekeelse sõna ja kuvab selle vaste eesti keeles, kui inglisekeelne sõna on sõnastikus olemas. Kasutatakse samu loendeid **eesti** ja **inglise**, mis eelmises näites.

Muutujad

n – elementide arv loendites,

k – elemendi järjenumbr (indeks),

sõna – sõna inglise keeles

Tõlkimine. Algoritm

$n = \text{inglise.pikkus}$

$k = 1$

loe sõna

kordus n korda

kui sõna = **inglise** (k) siis

kuva „See on ” & **eesti** (k)

stopp

lõpp kui

$k = k + 1$

lõpp kordus

kuva „Sellist sõna ei ole!”



Programm loeb inglisekeelse sõna ja omistab selle muutujale **sona**. Seda otsitakse loendist **inglise**, võrreldes elemente küsitud sõnaga alates esimesest ja võimalik, et lõpuni. Kui sõna leitakse, kuvatakse eestikeelne vaste ja skripti töö katkestatakse. Kui sõna ei leita, lõpeb kordus ja kuvatakse teade „Seda sõna sõnastikus ei ole!”

Näide demonstreerib loendi elementide järjestikust töötlemist võimaliku katkestusega. Tegemist on üsna levinud ülesandega: etteantud väärtuse otsimisega loendis (massiivis). Toodud lahendus ei ole eriti hea.

Dialoogid, vestlused ja koomiksid



Võrdlemisi sarnaselt eelmiste näidetega võib organiseerida mitmesuguseid dialooge, vestlusi jmt.

Ühes loendis on ühe tegelase jutt, teises partneri jutt. Vestluse sisu ja pikkus võib olla suvaline.

Toodud skeemi kohaselt mängib üks tegelastest siin nõ juhtrolli, kes alustab vestlust ja kelle skriptis on kordus loenduri (järjenumbri) muutmiseks, mida kasutatakse lausete valimiseks loenditest. Pärast esimese tegelase iga lause kuvamist saadetakse teade teisele tegelasele, kelle skriptis lastakse kuvada sama numbriga tekst tema loendist.

Jutuga võib siduda ka mitmesuguseid tegevusi.

Ülevaate saamiseks taolistest rakendustest võiks tutvuda projektidega [Kraps koolis](#) ja [Vestlus tiigriga](#)

Loendite elementide loomine programmiga

Enne kui skripti saab kasutada loendi elementide loomiseks, peab loend olema loodud käsuga **Tee loend** ning peab teada olema loendi nimi.

Loendi elementide sisestamine klaviatuurilt



Varem loodud loendisse saab elemente (nii arve kui tekste) sisestada klaviatuurilt.

Kõrval on toodud üks võimalikest skeemidest. Seda saab kohandada vastavalt oma rakenduse vajadustele. Eeskätt puudutab see loendi nime, mille võib välja vahetada.

Skript küsib elementide arvu **n**. Eemaldatakse loendi **V** kõik elemendid. Edasi järjest küsitatakse ja loetakse elemente. Igal küsimisel väljastatakse teade „Element **k**“, kus **k** on elemendi järjenumbr.

Loendi loomine juhuarvudest



Algoritmide kontrollimiseks on sageli vaja teatud andmete kogumeid. Neid saab mugavalt luua juhuarvudest.

Kõigepealt küsitatakse elementide arvu loendis ja omistatakse see muutujale **n**.

Seejärel küsitakse arvude minimaalne ja maksimaalne suurus; kasutaja antud väärtused omistatakse vastavalt muutujatele **min** ja **max**.

Korduses genereeritakse järjest juhuslikke arve vahemikus **min ... max** ning lisatakse need loendi lõppu.

Mõned tüüp algoritm loenditega

Loendi elementide summa ja aritmeetiline keskmine

[Demo](#)

Andmed

V – antud loend

n – elementide arv loendis

k – elemendi järjenumbr

Sum – elementide summa

kesk – aritmeetiline keskmine

Muutuja **Sum** algväärtuseks võetakse 0, muutuja **n** väärtuseks elementide arv loendis. Indeks **k** väärtust muudetakse järjest ühe võrra ja selle järjenumbriga element liidetakse summale.

protseduur Sum_kest

$Sum = 0$

$n = V.pikkus$

$k = 1$

kordus n korda

$S = S + V(k)$

$k = k + 1$

lõpp kordus

$kesk = Sum / n$



Maksimaalne element loendis ja selle järjenumbr

Andmed

V – antud loend

n – elementide arv loendis

k – järjenumbr

max – maksimaalne element

nr – maks. elemendi järjenumbr

Muutuja **max** algväärtuseks võetakse loendi esimene element, **nr** algväärtuseks on 1. Loendi elemente võrreldakse järjest **max** jooksva väärtusega; kui mõni element osutub suuremaks, võetakse see uueks maks väärtuseks ja jäetakse meelde ka järjenumbr.

protseduur Max_nr

$n = V.pikkus$

$max = V(1)$

$nr = 1$

$k = 1$

kordus n korda

kui $V(k) > max$ siis

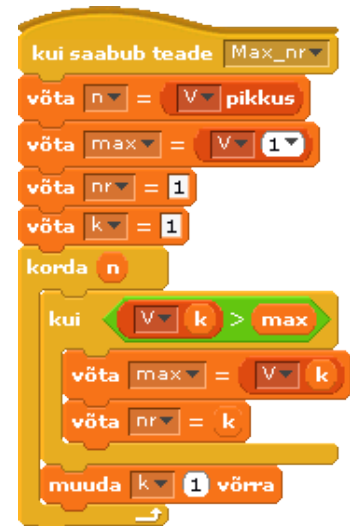
$max = V(k)$

$nr = k$

lõpp kui

$k = k + 1$

lõpp kordus



Loendite kasutamine tööks objektidega

[Näide](#). Autode võidusõit

Kui kasutusel on mitu ühetaolise käitumisega ühetüübilist objekti, siis on tavaliselt otstarbeks kasutada informatsiooni säilitamiseks loendeid. Antud juhul on võidusõidul osalemas viis autot. Põhimõtteliselt ei oma objektide arv tähtsust, kuid näiteks praegu võib tegelaste arvu piirata lava suurus.

	Sõitjad	Kohad	Ajad
1	Vettel	5 Massa	11.681
2	Shumi	2 Shumi	12.779
3	Button	4 Webber	13.7
4	Webber	3 Button	14.64
5	Massa	1 Vettel	15.656

+ pikkus: 5

Kasutusel on kolm loendit. Loendis **Sõitjad** on sõitjate nimed, mis on ühtlasi ka spraitide nimed. Seega on tegemist viitadega objektidele. Loendisse **Kohad** tekivad sõidu järel võistlejate nimed sõidu lõpetamise järjekorras ning loendis **Ajad** vastavad ajad.

Objektid: autod (5), Kraps

Loendid: Sõitjad, Kohad, Ajad

Muutuja: ringe

Protseduur Autosõit

auto.x = lava.minX

kordus ringe **korda**

kordus kuni auto.x > lava.maxX

liigu juhuarv(15...20) sammu

lõpp kordus

auto.peida

paus 1...3 sek

auto.x = lava.minX

auto.näita

lõpp kordus

lisa Kohad nr ja Sõitjad(nr)

lisa Ajad taimer



Enne sõidu algust kustutatakse loendite **Ajad** ja **Kohad** sisu (eelmise sõidu tulemused), nullitakse taimer ja käivitatakse käsuga [**teavita Start**] autode skriptid. Need on praktiliselt ühesugused. Erinevus on ainult plokis [**lisa Kohad ...**], kus iga auto jaoks on oma number: 1, 2, ... Siintoodu on 3. auto skript.

