

Projekt. Kraps, papagoi ja pall



[Demo](#) [Sissejuhatus Scratchi Kasutamisuhand](#)
[Lindistus Kraps, pall ja papagoi](#)

Võiks vaadata läbi lindistuse ja siis teha ise rakenduse toodud juhise järgi

Ülesande püstitus

Koostada mäng vastavuses järgmise stsenaariumiga ja demoga.

Kraps jalutab rahulikult **palliga** edasi-tagasi rannas. Järsku ilmub taevasse **papagoi**, kes lendab ringi ja narrib Krapsu.

Kraps üritab, mängija abiga, tabada palliga papagoid: palli lennutamiseks peab kasutaja vajutama **tühiku** klahvi.

Mängu eesmärgiks on tabada papagoid võimalikult rohkem kordi antud aja

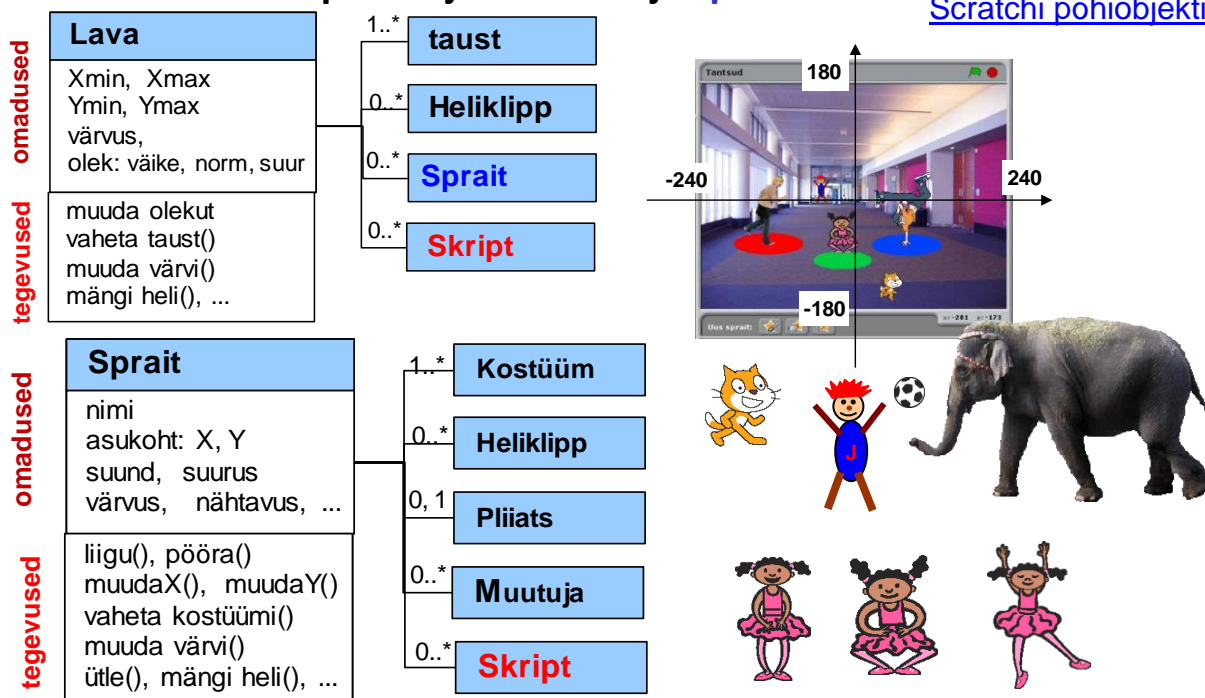
Projekti tegemise käigus tutvume või kinnistame

Scratch'i põhielementidega ja programmeerimise üldpõhimõtetega :

kasutajaliides, spraidid ja kostüümid, lava ja taustad, skriptid ja käsud, tegevuste määramine spraitidega: asukohta muutmise, pööramine, peitmine jm programmi olemus ja töö põhimõtted, objektid, nende omadused ja tegevused, protsesside juhtimine: kordused ja valikud, järjestikused ja paralleelsed protsessid, koostöö programmiüksuste vahel, andmete liigid, muutujate olemus ja kasutamine programmides

Scratchi põhiobjektid: lava ja spraidid

[modelleerimine Scratchi põhiobjektid](#)



Harjutus. Kraps, papagoi ja pall

[Demo](#)

[Kasutamisujuhend](#)

1. Krapsu liikumine edasi tagasi (ilma pallita)

Spraitide **põhiomadused**: nimi, asukoht, suurus,... ja **meetodid**: liigu, pööra, .. Skriptide loomine. Käsud ja käsuplokid, liht- ja liitkäsud. Taustade lisamine. Spraitide liikumine, kostüümide kasutamine, pauside tekitamine, animatsioon. Helide mängimine. Tekstide kuvamine. Protsesside juhtimine: jada ja kordus. Skriptide käivitamine ja peatamine. Klahvi ja hiire sündmused

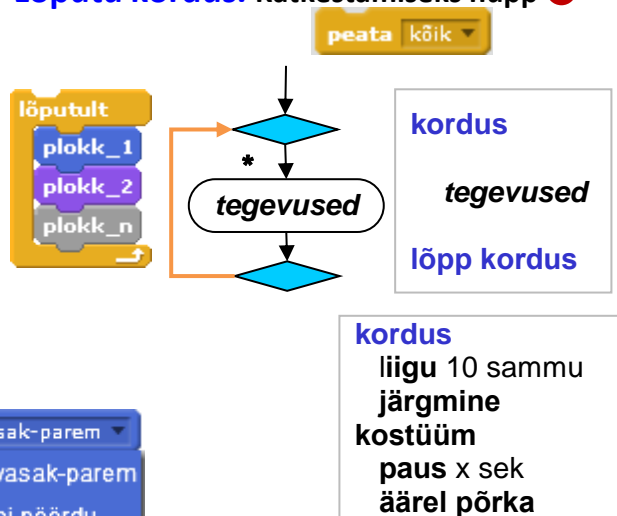
liitkäsud: liigu, järgmine kostüüm, oota

tingimuslik käsk: kui äärel pörka

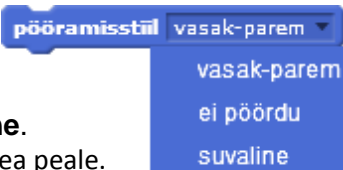
liitkäsud lõnutult: seesolevaid käskse täidetakse nõhimõtteliselt lõnutult



Lõputu kordus. Katkestamiseks nupp ●



kui äärel, pörka



Vaikimisi pööramisstiil **suvaline**.

Pörkamisel pööre 180°, Kraps pea peale.

Pööramisstiil **vasak-parem**: pööre ümber vertikaalse telje



Skripti **käivitamiseks**: a) klõpsata seda b) **roheline lipp**, c) käsk [teata nimi ...]

peatamiseks: a) klõps, b) **punane nupp** (kõik skriptid), c) käsk [peata ...]

Helid: valmis klipid: süsteemiga kaasas, omad **WAV** ja **MP3**-failid, saab lindistada.

Taustad: graafikafailid - süsteemiga kaasas, võib oma faile (GIF, JPEG, PNG, ...)

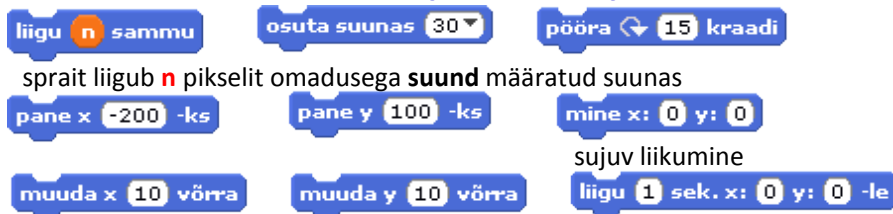
Lisamiseks: muuta **Lava** aktiivseks ja nupud **Uus taust**: Tausta valimine teegist, Tausta laadimine failist, ...

2. **Krapsu** pööramine ja liigutamine nooleklahvidega
Klahvisündmused, *Spritide liigutamine nooleklahvidega, spraidi suuna muutmise*

päiseplokk **kui vajutakse klahvi**, käsk **osuta suunas**
 Teha kaks skripti parema ja vasaku nooleklahvi jaoks



Liikumise käsud (meetodid)



Lugemiseks, vaatamiseks, uurimiseks, proovimiseks

[objektid](#) [skriptid](#) [juhtimine](#) [Sissejuhatus Scratchi](#)

Harjutus. Kraps, papagoi ja pall [Demo](#)

3. Papagoi "ringliiklus" ja "narrimine"

Spraitide ja kostüümide lisamine, skriptide kopeerimine ja redigeerimine, tingimused, valikud ja **kui**-plokid, spraitide asukoha määramine, spraitide peitmine ja kuvamine, spraitide „ringliiklus“, juhuslikute arvude kasutamine, paralleelprotsessid

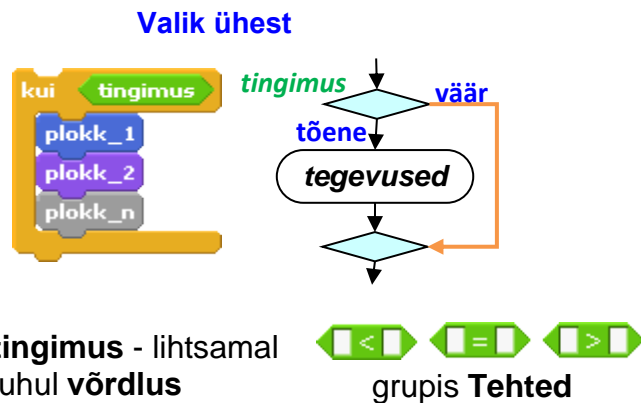
lihtkäsud: **peida**, **näita**, **üttele**, **pane x (pane y, mine x ... y ...)**

tingimuslik liitkask **kui**: tingimused, valik ühest ja kahendvalik

Lisage sprait **Parrot** ja pange sellele nimi **papagoi**. Nupp "**Spraidi valimine teegist**". Mitu kausta ja teemat: Loomad, Fantaasia, Inimesed, ... Spraidil on kaks kostüümi Kopeerige **Krapsu** skript **papagoile** ja proovige. Ringliikuse korraldamiseks muuta: kui sprait jõuab lava parema servani, peita (käsk **peida**) korra (1-3 sek) sprait, viia vasakusse serva ja tuua peidust välja (**näita**). Pausi tekitamiseks kasutada **juhuslike arve**.

Et teha tabamine raskemas, võiks käsu **pane x -240** asemel võtta näiteks käsu **pane x** juhuarv -240 kuni 120. Papagoi ilmub nüüd nähtavale juhuslikus kohas antud vahemikus. Proovige!

Põhiskriptid - ringliiklus



Võiks lisada algusesse toodud käsud, a) et papagoi ilmiks juhuslikul momendil b) et oleks tagatud papagoi liikumine vasakult paremale

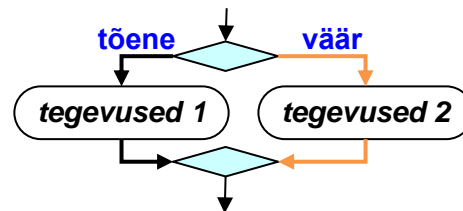


Lisada **papagoi** "jutt" (narrimine): **kui** asukoht < 0-st **jutt1** **muidu** **jutt2**

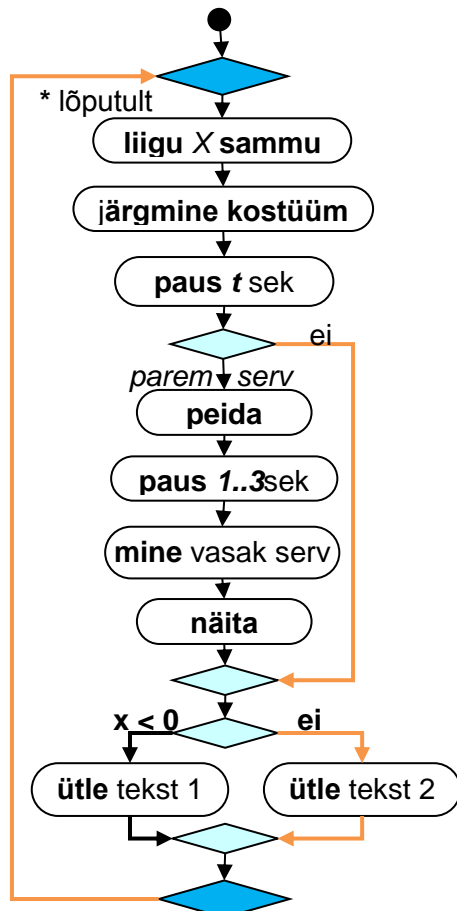
Narrimine



Kahendvalik



Algoritm - UML' tegevusskeem



Algoritm - pseudokood (algoritmikeel)

```

korda lõputult
  liigu X sammu
  järgmine kostüüm
  paus t sek
  kui parem serv siis
    peida
    paus 1..3 sek
    mine vasak serv
    näita
  lõpp kui
  kui x < 0 siis
    üttele tekst1
  muidu
    üttele tekst2
  lõpp kordus
  
```

Lugemiseks, vaatamiseks, uurimiseks, proovimiseks

[juhtimine](#) [algoritm](#) [modelleerimine](#)

Harjutus. Kraps, papagoi ja pall

[Demo](#)

4. Palli lisamine: uue spraidi tegemine **joonistamisredaktoriga**

joonistamisredaktori käivitamine, redaktori põhivõimalused ja tööriistad

Redaktori käivitamiseks **Uus sprait**, nupp "**Uue spraidi joonistamine**"

Näiteks teha ring, panna sellele suvalised täpid ja laigud vmt



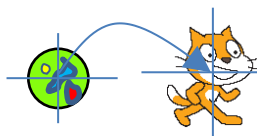
5. Palli liikumine **Krapsu** juurde ja sealt lava ülemise servani

täiendavad klahvi sündmused: skripti käivitamine vajutusega klahvile, sündmus "puudutab sev", ühe objekti viimine teise juurde, spraidi keskpunkt (nullpunkt), tingimuslik kordus

lihtkäsud: **mine** ..., **muuda** y ... (muuda x ...)

liitkäsk: **korda kuni tingimus** (saab tõeseks)

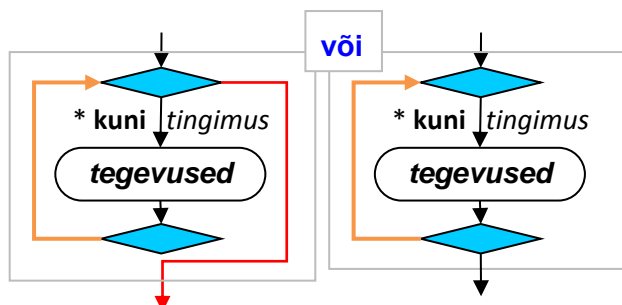
mine Kraps
näita
korda kuni *puudutab serv*



Käsk [**mine** sprait] paneb kohakuti nullpunktid. Spraidi (täpsemalt, kostüümi) nullpunkt asub vaikimisi selle pinnakeskmes.



Eelkontrolliga tingimuslik kordus.



tegevusi korratakse seni, kuni tingimus saab tõeseks

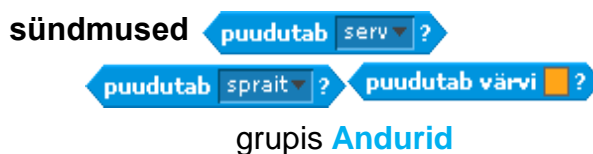


korda kuni *tingimus*

tegevused

lõpp kordus

[juhtimine](#)



6. Palli kokkupuude papagoiga

sündmus ja tingimus **on puude**, skripti peatamine, teadete saatmine ja vastuvõtmine

käsud: **peata skript**, **teavita ...** ning **teavita ... ja oota**

mine Krops
näita
kordus kuni **serv**
muuda y h võrra
kui **puudutab papagoi**
peida
teata **Pihtas**
peata skript
lõp kui
lõpp kordus
peida
teata **Möödas**



Alternatiiv!

Papagoi puudutab palli!

Peab papagoi skriptis

Teated teisiti.

Ressurside "raiskamine" - kontroll peab kogu aeg, ka siis kui pall ei lenda!

7. Teadete saatmine ja vastuvõtmine. Spraitide suhtlemine

Sündmus ja päiseplokki **kui saabub teade**

Teated **Pihtas** ja **Möödas** võtavad vastu **papagoi** ja **Krops** ja käivitavad vastavad skriptid. Põhimõtteliselt võib teate võtta vastu suvaline skript, mis algab päiseplokiga **kui saabub teade nimi**.

NB! teata ... saatja töö jätkub **kohe**

teata ... ja **oota** saatja töö jätkub peale seda, **kui** vastuvõtjad **täidetud**

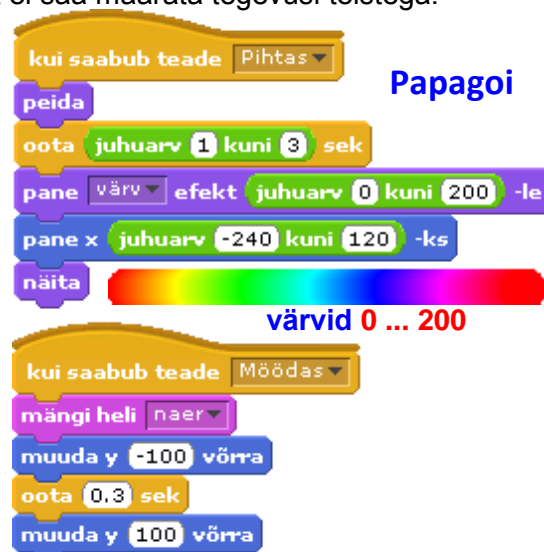
NB! Milleks? Praegu sellepärast, et üks sprait ei saa määrata tegevusi teistega!

Pihtas: **papagoi** korraks peidetakse,

Krops ütleb midagi, ...

Möödas: **papagoi:** naer, alla-ülesse?,

Krops ... graafika efektid: keere jm



Harjutus. Kraps, papagoi ja pall

[Demo](#)
[Muutujad](#)

8. Visete ja tabamuste loendamine. **Muutujad!!!**

Muutuja - nimega varustatud **koht** (*pesa, väli*) arvuti mälus - **objekt**. Muutujasse saab programm **salvestada** väärtusi (õeldakse ka **omistada** väärtusi). Igal ajahetkel saab muutujal olla ainult **üks väärtus**. Salvestatud väärtust saab programm kasutada (**lugeda**) näiteks uue väärtuse leidmiseks.

Scratch'is saab muutujaid luua projekti loomise faasis grupis **Andmed** oleva korraldustga **Loo muutuja**. Skriptides saab muutujatele **omistada** väärtusi ja **muuta** väärtusi käskudega **võta** ja **muuda**:

võta viskeid = 0 **muuda** tabas 1 võrra

Muutuja jooksvat väärtust saab lasta kuvada laval nn **monitoris**. Märkeruut!

Luu kaks muutujat: **viskeid** ja **tabas**



NB! globaalsed (kõik spraidid ja lava) ja **lokaalsed** (ainult üks sprait) muutujad

Teeme **globaalsed** muutujad: **viskeid**, **tabas**

Milline sprait muudab? Erinevad võimalused. Praegu võiks **pall** või **lava**. Viimasel Kui inimene **saab aru** mis on muutuja, siis ta juba **oskab programmeerida!**

Niclaus Wirth programmeerimiskeelte Pascal. Modula, Oberon jt looja, paljude õpikute, raamatute ja artiklite autor

Igale **muutujale** eraldatakse

tabas	2
viskeid	5
...	

Muutuja

nimi
väärtus
skoop

lisa()
eemalda()
salvesta()
loe()

NB! NB! NB! Muutujate algväärtustamine!

Korduval käivitamisel, jätkatakse eelmisest väärtusest.

Peaks uue mängu alguses vastavad käsud. **pall** või **lava**.

[muutujad](#) [rakendused](#)

Harjutus. Kraps, papagoi ja pall

[Demo](#)

9. Aja näitamine ja kasutamine töö lõpetamiseks

*Taimer kasutamine: käsk **taimer algseisu**. Käsk **peata kõik**.*

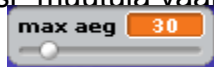
Liuguri (kerimisriba) kasutamine muutujate väärtuste määramiseks

Teha kaks muutujat: **aeg** (jooksev aeg) ja **max_aeg** (mängu aeg)

Muutuja monitorile saab lisada **liuguri** (kerimisriba), mille abil

saab muuta "käsitsi" muutuja väärtust. Teeme seda muutuja

max_aeg jaoks



Teeme skript aja näitamiseks ja mängu lõpetamiseks aja järgi. Taimer käsud asuvad grupis **Andurid**

kui klõpsatakse **rohelist lippu**

taimer algseisu

korda kuni aeg > max_aeg

aeg = taimer

peata kõik

5.7

Aja kuvamiseks võiks kasutada monitori suure näidikuga

Skripti võib panna suvalise spraidi moodulisse: näiteks **lava** või Kraps.

Tüüpiliselt taolised üldisega iseloomuga skriptid paigutakse **lava** moodulisse.



10. Töö lõpetamine visete arvu järgi

Käsk [oota kuni tingimus].

Teha muutuja: **m_viskeid**: visete maksimaalne arv

Skripti võib panna suvalise spraidi moodulisse: näiteks lava või Kraps

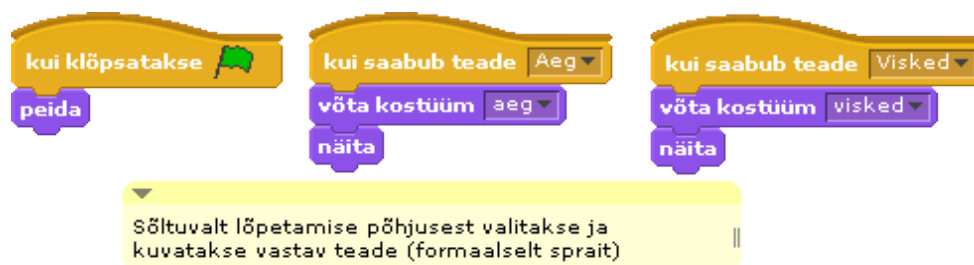


11. Sprait "teated" - "tekstispraidid"

Teadete, selgituste jmt kuvamiseks võib ploki **üttele** asemel kasutada nõ "tekstisprait". Spraidil võib olla mitu kostüümi: igaüks neist kujutab omaette teadet. Vastavalt olukorrale võetakse sobiv kostüüm ja kuvatakse sprait.

Praegu on kasutusel kaks teadet:

Aeg on läbi! (kostüüm Aeg)



Harjutus. Kraps, papagoi ja pall (järg)

12. Palli liikumine koos Krapsuga

Spraitide koostöö, teise spraidi omaduste (asukoht, suund jm) teadasaamine ja kasutamine, spraidi nihutamine ja pööramine

Paneme palli liikuma koos Krapsuga. Pall võiks ka pöörelda. Ei ole triviaalne. Liikumiste sünkroniseerimine! Võib erinevalt

Teeme skripti **Tule**, mis viib palli **Krapsu** juurde, pöörab ja paigutab selle, arvestades Krapsu liikumise suunda.

Lisada teavitamise käsk **Krapsu** põhiskripti.

Skript viib palli Krapsu juurde, kui see saadab teate Tule

```

kui saabub teade Tule
mine Krapsu juurde
muuda y -20 võrra
kui suund kujundil Kraps = 90
  pööra 30 kraadi
  muuda x 25 võrra
muidu
  pööra -30 kraadi
  muuda x -25 võrra

```

The script consists of several blocks: a 'when received Tule' block, a 'go to Kraps' block, a 'change y by -20' block, and a conditional 'if' block. The 'if' block checks the 'direction' of the 'Kraps' sprite. If it is 90 degrees, it rotates the ball 30 degrees clockwise and moves it 25 units to the right. Otherwise, it rotates the ball 30 degrees counter-clockwise and moves it 25 units to the left. Illustrations of the cat and ball are included.

```

kui klõpsatakse
  mängi heli näu
  ütle Tere! Olen Kraps! 2 sekundit
  lõputult
    liigu 10 sammu
    järgmine kostüüm
    oota 0.1 sek
    kui äärel, pörka
    teavita Tule ja oota

```

The script starts with a 'when clicked' block, followed by 'play sound: clap', 'say: Hello! I am Kraps! for 2 seconds', and a 'forever' loop containing 'move 10 steps', 'next costume', 'wait 0.1 seconds', 'if on edge, bounce', and 'send Tule and wait'.

13. Palli liikumine koos Krapsuga ja lendamine.

Olekumuutujad

Mis juhtub, kui pall pannakse lendama?

Tuleb tagasi!

Võtta kasutusele nn **olekumuutuja** (tunnus) **lendab**:
kui **pall** lendab on muutuja väärtus **1**,
vastupidisel juhul **0**

Muutuja väärtust muudab **palli** põhiskript

Algväärtus 0, koos teiste muutujatega

Muuta **Krapsu** skripti, arvestamiseks, kas pall
lendab või mitte

[muutujad](#) [skriptid](#)

14. Palli põhiskripti ja algväärtustamise lõppvariant

Palli põhiskript. Juhib palli lendu. Lennu lõpetamiseks kaks võimalust:
 1) pall tabab papagoid; teade Pihtas 2) pall jõuab servani; teade Möödas
 Skript loendab visete ja tabamuste arvu ja muudab muutuja lendab väärtust

kui klõpsatakse
 võta viskeid = 0
 võta tabas = 0
 võta lendab = 0

Muutujate algväärtustamine
 viskeid - visete arv
 tabas - tabamuste arv
 lendab - palli olekumuutuja
 1 - lendab, 0 - ei lenda

kui vajutatakse klahvi tühik
 mine Krops
 muuda viskeid 1 võrra
 võta lendab = 1
 Pall Kropsu juurd

kui klõpsatakse
 võta viskeid = 0
 võta tabas = 0
 võta lendab = 0
 Pall lendu

korda kuni puudutab serv ?
 muuda y 20 võrra
 oota 0.01 sek
 kui puudutab papagoi ?
 muuda tabas 1 võrra
 võta lendab = 0
 Teate võtavad vastu papagoi ja Krops
 Kask peatab antud skripti töö
 teavita Pihtas
 peata skript
 võta lendab = 0
 Palli lend on lõpenud. Jõudis servani
 Teate võtavad vastu papagoi ja Krops
 teavita Möödas

Tingimuslik kordus
 Seesolevaid käskke täidetakse seni kuni tingimus saab tõseks: pall puudutab serva. Kordamised lõpetakse ja skripti töö katkestatakse, kui pall puudutab papagoid

15. Kropsu põhiskripti lõppvariandid

Krops jääb seisma kui pall lendab. Tingimus Pall liigub Kropsuga, kui ei lenda (**lendab** = 0)

lendab = 0 kehtib kogu liikumise kohta

kui klõpsatakse
 mängi heli näu
 ütle Tere! Olen Krops! 2 sekundit
 osuta suunas 90
 korda lõputult
 kui lendab = 0 siis
 liigu 10 punkti
 järgmine kostüüm
 oota 0.2 sek
 kui äärel, pörka
 teata Tule ja oota
 pall Kropsu juurde

käskke täidetakse, kui muutuja lendab = 0. Kui lendab = 1, Krops ei liigu, pall liigub vertikaalselt

Lisaülesanded

- Kraps teeb salto, kui tabab papagoi
- kuvatakse palli lend tagasi
- arvutakse ja kuvatakse tabamusprotsent
- omapoolsed lisategevused

[Demo](#)

[Sissejuhatus Scratchi](#)

[Lindistus Kraps, pall ja papagoi](#)