Cikls ar skaitītāju

# Cikls

Cikls (looping) ir vadības konstrukcija, kas atkārto noteiktas darbības izpildi līdz brīdim, kad pārstāj izpildīties noteikts nosacījums

Programmēšanā mēdz izšķirt divu veidu ciklus:

cikls ar skaitītāju,

cikls ar nosacījumu (conditional looping).

Ciklā ar skaitītāju pirms cikla izpildes jau ir zināms atkārtošanas reižu skaits.

# Cikla FOR priekšraksts

Šo cikla priekšrakstu lieto tad, ja ir zināms cikla atkārtošanās reižu skaits. To sauc par ciklu ar skaitītāju.

Veidojot ciklu FOR tiek noteikta mainīgā sākuma vērtība, beigu nosacījums un cikla mainīgā vērtības izmaiņas nosacījums. Visi parametri ir atdalīti ar semikolu

for (*cikla mainīgais* =sākuma vērtība; beigu nosacījums; cikla mainīgā vērtības izmaiņa){

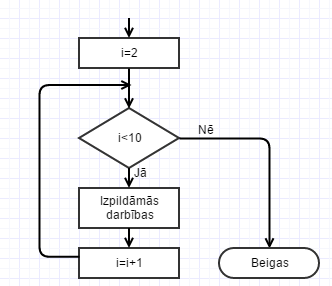
<*izpildāmās darbības*>

}

Piemērs

for (var i =2; i<10; i=i+1){

document.write(i);

}

i – cikla mainīgais, jeb skaitītājs ar sākuma vērtību 2;

i<10 – nosacījums, kuram neizpildoties cikls tiek pārtraukts;

i=i+1 – izteiksme, kas norāda cikla mainīgā vērtības izmaiņu, šajā gadījumā i vērtība atkārtojoties ciklam, tiek palielināta par 1;

Programma beidz izpildīt ciklu tad, kad nosacījums i<10 kļuvis aplams (piemēram, pārsniedz 10)

J*avaScript* valodā i=i+1 var aizstāt ar i++ un i=i-1 var aizstāt ar i--

# Piemēri

|  |  |
| --- | --- |
| for (var i =2; i<10; i=i+1) {  document.write(“i”);  } |  |
| for (var i =2; i<10; i++) {  document.write(i+ “ “);  } |  |
| for (var i =2; i<10; i++) {  document.write(i+ ‘<br>’);  } |  |
| for (var i =2; i<10; i=i+2) {  document.write(i+ “ “);  } |  |
| var beigas=10;  for (var i =1; i<beigas; i=i+2) {  document.write(i+ “ “);  } |  |
| var sakums=1;  var beigas=10;  for (var i=sakums; i<beigas; i++){  document.write(i+ " ");  }  document.write('<br>');  document.write(" Izvadīju visus skaiļus no " + sakums + "līdz " +(beigas-1)); |  |
| var sakums=2;  var beigas=10;  for (var i=sakums; i<=beigas; i++){  document.write(i +" kvadrātā ir "+i\*i+'<br>');  } |  |

# Riņķa vai riņķa daļas zīmēšana

Riņķi vai riņķa daļas var veidot ar arc metodi.

*JavaScrip* arc metodes pieraksts *arc(x, y, r, Angle1, Angle2), kur*

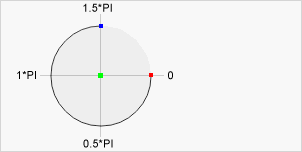
x – riņķa centra x koordināte;

y – riņķa centra y koordināte;

r – rinķa rādiuss;

Angle1 – sākuma leņķis (radiānos);

Angle2 – beigu leņķis (radiānos).



• Centrs arc (**100,75**, 50,0,1.5\*Math.PI)

• Sākuma leņķis arc(100,75, 50,**0**,1.5\*Math.PI)

• Beigu leņķis arc(100,75, 50,0,**1.5\*Math.PI**)

Strādā tikai pamatnē (canvas).

var c=document.getElementById("myCanvas");  
var ctx=c.getContext("2d");  
ctx.beginPath();  
ctx.arc(100,75,50,0,2\*Math.PI);  
ctx.stroke();

# Gadījuma skaitļu ģenerēšana

Gadījuma skaitļu ģenerēšana *JavaScript* notiek ar metodi Random(), kur skaitļi tiek izvēlēti intervālā starp [0 un 1) Math.random().

Ja nepieciešams gadījuma skaitļus ģenerēt intervālā [1;10], tad priekšraksts varētu būt šāds

Math.floor((Math.random() \* 10) + 1)