

Astotā nodaļa

Iepriekšējā nodaļā noskaidrojām, ka procesu, kad kāda darbība tiek veikta vairākkārtīgi, sauc par ciklu, bet lielumus, kuru vērtības mainās ciklā - par cikla mainīgajiem. Apskatījām *for* cikla konstrukciju. Kā noskaidrojām, cikla operatoru *for* var izmantot, ja iepriekš ir zināms, cik reizes cikls atkārtosies. Bet, ko darīt, ja tas iepriekš nav paredzams? Šādiem gadījumiem Java ir paredzētas *while* un *do...while* cikla konstrukcijas. Šajā nodaļā uzzināsim par *while* cikla operatoru un tā piedāvātajām iespējām. Savukārt ar *do...while* operatoru iepazīsimies nākamajā nodaļā.

8.1. Cikla operators *while*

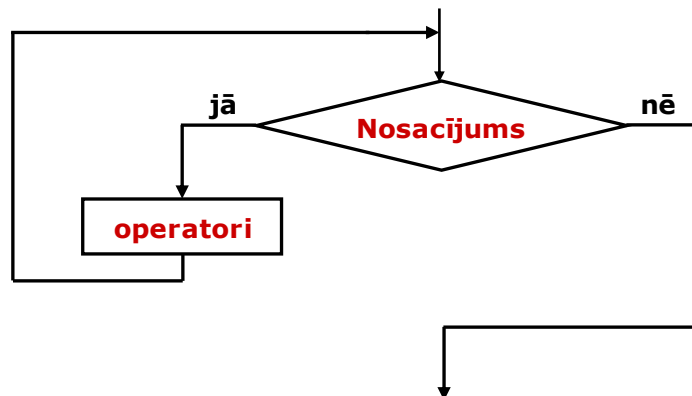
While ir cikla konstrukcija ar nosacījumu cikla sākumā. Tas nozīmē, pirms uzsākt cikla darbību jeb cikla ķermeņa operatoru izpildi tiek pārbaudīts cikla sākuma nosacījums. Ja nosacījums ir patiess, tiek uzsākta cikla darbība, bet, ja nosacījums ir nepatiess, cikls netiek pildīts.

Vispārīgā pieraksta forma:

```
while (<nosacījums>)
{
    <izpildāmie operatori>;
}
```

Cikls izpilda cikla operatorus, kamēr ir spēkā nosacījums. Jāievēro, ka cikla parametram pirms cikla jāpiešķir kāda vērtība, un vienam no cikla operatoriem šī vērtība ir jāizmaina, jo citādi cikls turpināsies bezgalīgi.

Konstrukcijas *while* blokshēma:



Piemēram, sastādīsim programmu, kas ļauj noskaidrot cik ilgi jākrāj nauda, ja vēlas sakrāt N eiro, un zināms, ka pirmajā dienā krājkasē ieliek X centus, bet katrā nākošajā dienā – divas reizes vairāk nekā iepriekšējā.

Uzd_8_1.java	
1.	<code>import java.util.Scanner;</code>
2.	<code>public class Uzd_8_1 {</code>
3.	<code> public static void main(String[] args) {</code>
4.	<code> Scanner dati=new Scanner(System.in);</code>
5.	<code> double N; //vēlamā summa EUR</code>
6.	<code> int n; //vēlamā summa centos</code>
7.	<code> int X; //cik centus ieliek krājkasē pirmajā dienā</code>
8.	<code> int K=0; //cik centi ir uzkrāti kopā</code>
9.	<code> int D=0; //cik dienas jākrāj</code>
10.	<code> System.out.print("Cik EUR vēlaties sakrāt: ");</code>
11.	<code> N=dati.nextDouble();</code>
12.	<code> System.out.print("Cik centus liksiet krājkasē pirmajā dienā: ");</code>
13.	<code> X=dati.nextInt();</code>
14.	<code> n=(int)N*100; //pārrēķina vēlamo summu no EUR uz centiem</code>

15.	<code>//cikls, kas strādās tik ilgi, kamēr nebūs uzkrāta vēlāmā summa</code>
16.	<code>while(K<n){</code>
17.	<code> D=D+1; //dienu skaitu palielina par 1</code>
18.	<code> K=K+X; //palielina uzkrājumu</code>
19.	<code> X=X*2; //dubulto centu skaitu, kas jāliek krājkasē</code>
20.	<code> }</code>
21.	<code> //pēc cikla izvada paziņojumu, cik dienas jākrāj nauda</code>
22.	<code> System.out.println("Naudu nāksies krāt "+D+" dienas");</code>
23.	<code> dati.close();</code>
24.	<code>}</code>
Programmas darbības rezultāts: piemēram, ja lietotājs vēlas sakrāt 12,50 EUR un krāšanu uzsāks ar 5 centiem, tad programmas paziņos, ka nauda jākrāj 8 dienas.	
Cik EUR vēlaties sakrāt: 12,50 Cik centus liksiet krājkasē pirmajā dienā: 5 Naudu nāksies krāt 8 dienas	

Programmas piemēra analīze:

Paskaidrojums	<i>N</i>	<i>n</i>	<i>D</i>	<i>K</i>	<i>X</i>	Ekrāns
Uzdotā sākuma vērtība			0	0		
Lietotāja ievadītie dati	12,50				5	
Nauda pārrēķināta centos		1250				
<i>K</i> ir mazāks par <i>n</i> , darbu sāk cikls			1	5	10	
<i>K</i> ir mazāks par <i>n</i> , cikls atkārtojas			2	15	20	
<i>K</i> ir mazāks par <i>n</i> , cikls atkārtojas			3	35	40	
<i>K</i> ir mazāks par <i>n</i> , cikls atkārtojas			4	75	80	
<i>K</i> ir mazāks par <i>n</i> , cikls atkārtojas			5	155	160	
<i>K</i> ir mazāks par <i>n</i> , cikls atkārtojas			6	315	320	
<i>K</i> ir mazāks par <i>n</i> , cikls atkārtojas			7	635	640	
<i>K</i> ir mazāks par <i>n</i> , cikls atkārtojas			8	1275	1280	
<i>K</i> nav mazāks par <i>n</i> , cikls darbu beidz						D = 8

Piebilde, ja ciklā jāizpilda tikai viens operators, figūriekavas - "{" un "}" var nelietot. Bet, ja ciklā jāizpilda vairākas komandas, tad tās obligāti jāraksta starp figūriekavām "{" un "}".

8.2. Cikla izmantošana darbam ar datnēm

Septītajā nodaļā aplūkojām, kā varam nolasīt datus no teksta datnes, ja iepriekš zināms, cik ierakstu satur teksta datne. Tomēr, ne vienmēr ir zināms, cik ierakstus satur datne, tādēļ aplūkosim piemēru, kā izmantojot ciklu ar priekšnosacījumu varam nolasīt datus no datnes, ja iepriekš nav zināms, cik ierakstus satur datne.

```

1.  import java.io.*;
2.  public class Uzd_8_2 {
3.      public static void main(String[] args) {
4.          try{
5.              //atver datni skaitli.txt un sagatavojas tās nolasīšanai
6.              BufferedReader lasa=
7.                  new BufferedReader(new FileReader("skaitli.txt"));
8.              //pagaidu mainīgais vienas teksta rindas nolasīšanai
9.              String pag;
10.             int x;
11.             pag=lasa.readLine(); //nolasa no datnes pirmo rindu
12.             //kamēr no datnes nolasītā rinda nav tukša
13.             while(pag!=null){
14.                 //tekstu konvertē par skaitli
15.                 x=Integer.parseInt(pag);
16.                 System.out.print(x+" "); //izvada skaitli ekrānā
17.                 pag=lasa.readLine(); //nolasa nākošo datnes rindu
18.             }
19.             //aizver datni
20.             lasa.close();
21.         } catch (IOException e){
22.             System.out.println("Problēmas ar datni");
23.         }
24.     }
25. }
    
```

Programmas darbības rezultāts: tiek nolasīti un izdrukāti ekrānā visi datnē "skaitli.txt" esošie skaitļi, neatkarīgi no tā, cik datnē to ir.

7; 10; -3; 8; 14;

8.3. Uzdevumi patstāvīgajam darbam

1. uzdevums

Uzzīmēt blokshēmu un sastādīt programmu, kas aprēķina, cik stāvi būs piramīdai, ja zināms, ka tā būvēta no kvadrātveida plāksnēm tā, ka augšējais stāvs sastāv no vienas plāksnes, otrais stāvs no augšas sastāv no 2×2 , jeb 4 plāksnēm, trešais stāvs no augšas sastāv no 3×3 , jeb 9 plāksnēm utt. Plākšņu skaitu ievada lietotājs.

2. uzdevums

Uzzīmēt blokshēmu un sastādīt programmu, kas nosaka, cik ciparu skaitli ievadījis lietotājs.

3. uzdevums

Sastādīt programmu, kas nolasa un izvada ekrānā datus no teksta datnes "draugi.txt", kurā ierakstīti draugu vārdi. Katrs vārds ierakstīts atsevišķā rindā.