

CIKLS AR SĀKUMA UN BEIGU NOSACĪJUMIEM

13.nodarbība

IEPRIEKŠ

FOR cikls

1. NO cik
2. LĪDZ cik
3. Ar kādu soli skaitīt

AR FOR CIKLU NEPIETIEK

- Vai robots zina, cik tālu tam ir jābrauc līdz konkrētai vietai?
- Paroles ievadišana dažādās lietotnēs.
- Spēļu noteikumi, lai uzvarētu vai tiktu nākamajā līmenī.
- Programma, kas «gaida», kamēr tiks nospiests kāds konkrēts taustiņš.

CIKLU IZVĒLE

Cikls ar skaitītāju

FOR

Cikls ar sākuma nosacījumu

WHILE

Cikls ar beigu nosacījumu

DO-WHILE

CIKLS AR SĀKUMA NOSACĪJUMU

```
while(nosacījums){
```

```
    izpildāmās darbības;
```

```
}
```

Kamēr nosacījums izpildās, veikt darbības.

CIKLS AR BEIGU NOSACĪJUMU

```
do{
    izpildāmās darbības;
}while(nosacījums);
```

Veikt darbības, kamēr izpildās nosacījums.

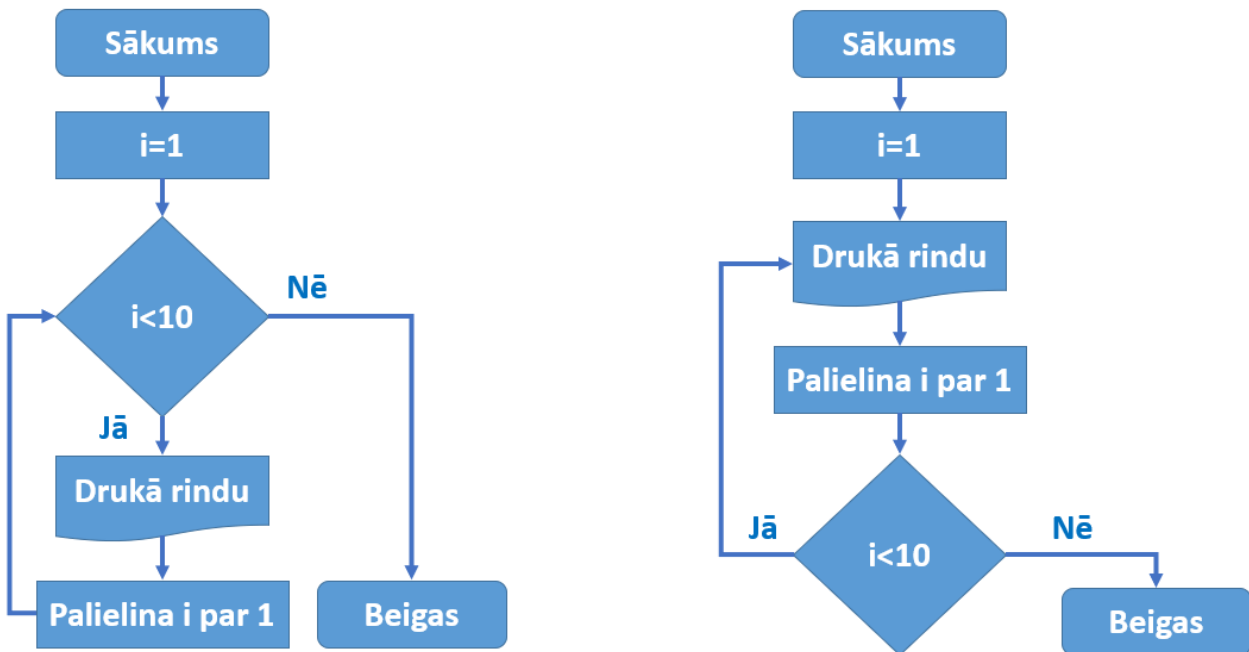
CIKLS AR SĀKUMA NOSACĪJUMU. SINTAKSE

```
package lv.startit.sample;
public class CiklsWhile {
    public static void main(String[] args); {
        int i=1;
        while (i<10){
            System.out.println(i);
            i++;
        }
    }
}
```

CIKLS AR BEIGU NOSACĪJUMU. SINTAKSE

```
package lv.startit.sample;
public class CiklsDoWhile {
    public static void main(String[] args); {
        int i=1;
        do {
            System.out.println(i);
            i++;
        }
        while (1<10);
    }
}
```

BLOKSHĒMA



UZDEVUMS

Kas notiks ar iepriekš minēto programmu, ja `i++` tiks izdzēsts?

PROGRAMMAS ANALĪZE

Programma, kas «vaicā» lietotājam ievadīt paroli. Ja parole ir pareiza, tad parāda paziņojumu «Parole ir pareiza!». Ja parole ir nepareiza, programma turpina «vaicāt» ievadīt paroli.

Programmā tiks izmantots Do-While cikls (jājautā tik ilgi, kamēr ievadītā parole ir pareiza).

PROGRAMMAS ANALĪZE. PROGRAMMA «PAROLE» I

```

package lv.startit.sample;
public class DoWhilePassword {
    public static void main(String[] args); {
        do {

```

```
        System.out.println("Ievadi paroli:");
    }
    while ( . . . );
    System.out.println("Parole ir pareiza!");
}
}
```

PROGRAMMAS ANALĪZE. PROGRAMMA «PAROLE» II

```
package lv.startit.sample;
import java.util.Scanner;
public class DoWhilePassword {
    public static void main(String[] args); {
        String password;
        do {
            System.out.println("Ievadi paroli:");
            Scanner scan = new Scanner(System.in);
            password = scan.nextLine();
        }
        while ( . . . );
        System.out.println("Parole ir pareiza!");
    }
}
```

PROGRAMMAS ANALĪZE. PROGRAMMA «PAROLE» III

```
package lv.startit.sample;
import java.util.Scanner;
public class DoWhilePassword {
    public static void main(String[] args); {
        String password;
        do {
            System.out.println("Ievadi paroli:");
            Scanner scan = new Scanner(System.in);
```

```
        password = scan.nextLine();
    }
    while (!password.equals("parole"));
    System.out.println("Parole ir pareiza!");
}
}
```

KĀ VARĒTU UZLABOT PROGRAMMU?

1. Pareizo paroli definēt ar mainīgo.
2. Uzskaitīt paroles ievades mēģinājuma reizes.
3. Pievienot papildus nosacījumus, kas jāizpilda, ja sasniegts maksimālais paroles ievades mēģinājuma reižu skaits.