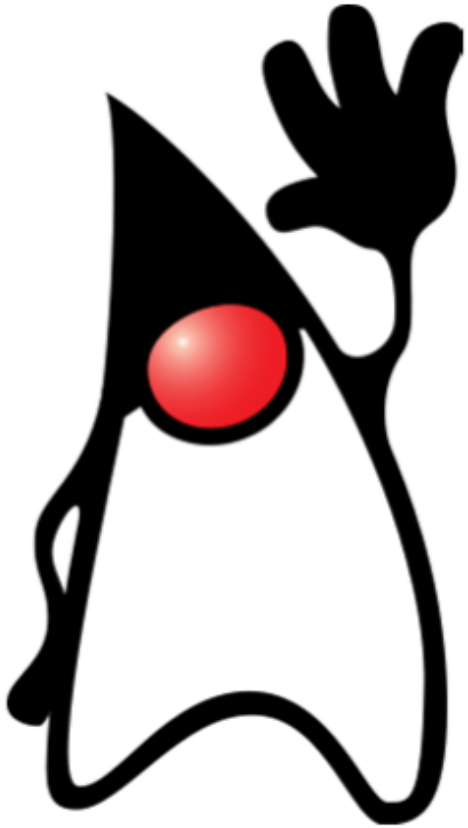
Programming Fundamentals
Lussen : While en Do-While

Klasgroep	1EO-ICT
Opleiding	Bachelor Elektronica-ICT
Theorie	DI: 8:45 - 9:45
	WOE: 11:45 - 12:45
(Werk)Labo	DI: 9:45 - 12:45 + 13:30 - 14:30
	WOE: 13:30 - 15:30 + 15:30 - 17:30
Docent	Katja Verbeeck
Contact	katja.verbeeck@odisee.be

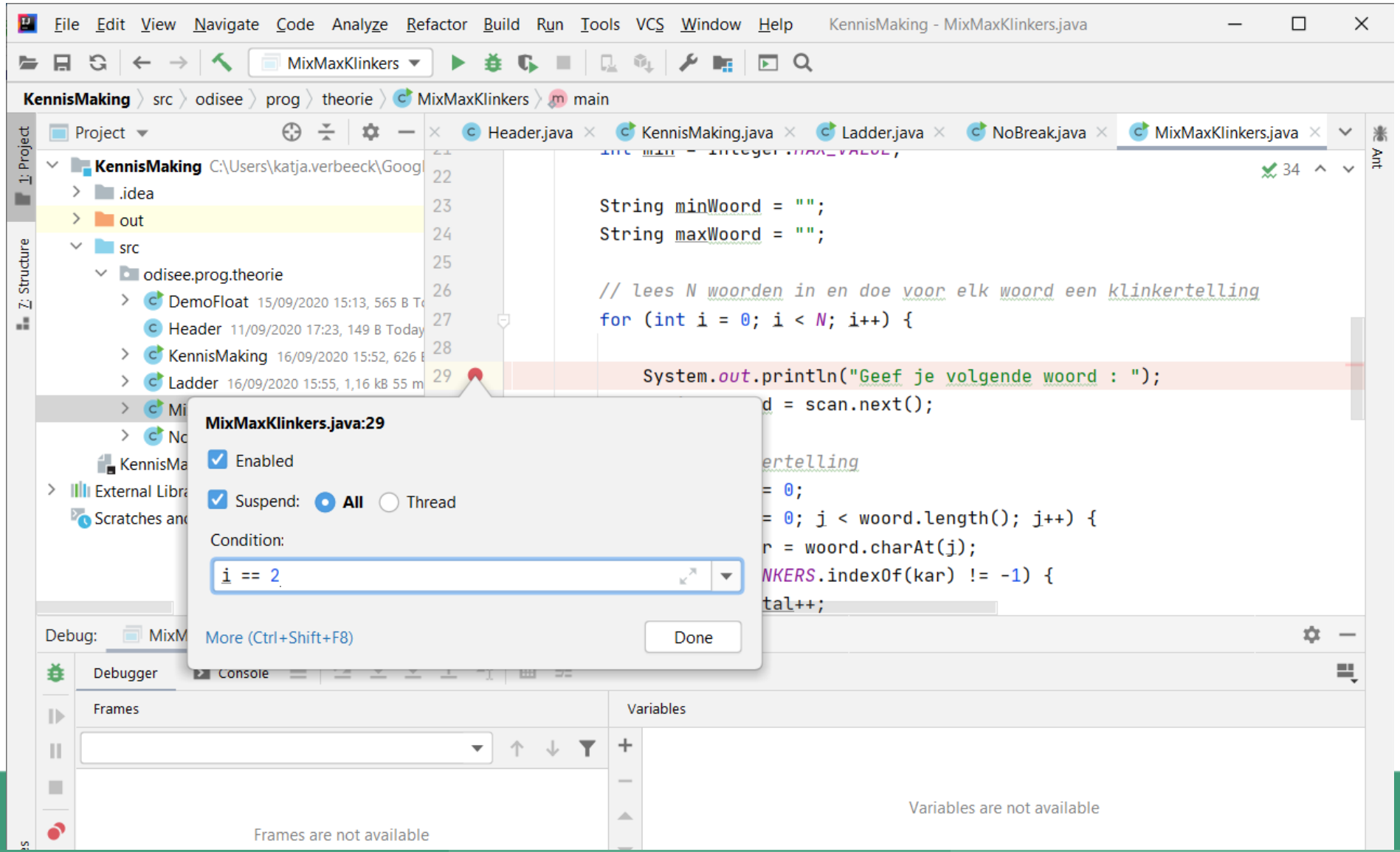
Inhoud



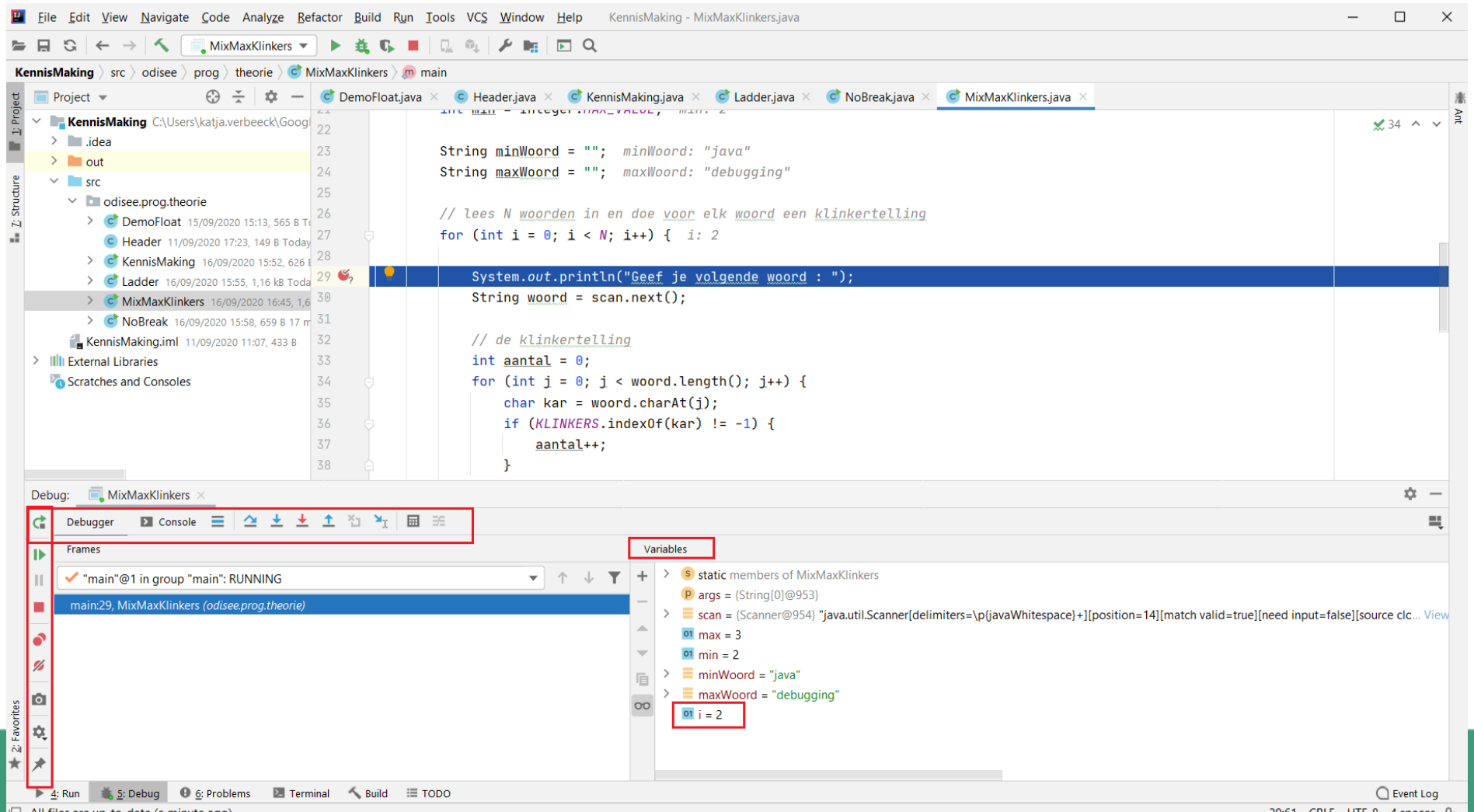
Oefening : MixMaxKlinkers

Schrijf een Java programma waarbij de gebruiker N woorden intypt en het woord met het minst aantal klinkers en het woord met het meest aantal klinkers opnieuw uitprint naar het scherm. Bij ex aequo print je het eerste woord af. Het getal N definieer je als constante.
$\rightarrow$ Hier zal een lus in een lus een oplossing geven $\rightarrow$ zie demo

Debugging : Debugpoints



Debugging : Variables en code step through



Algemene vorm van een while lus

Herhaal zolang een voorwaarde voldaan is. De voorwaarde wordt eerst gecheckt, de body van de lus wordt dus misschien nooit uitgevoerd !

Algemene vorm van een `while`-lus:

```
while (booleaanse uitdrukking) {
    ...
}
```

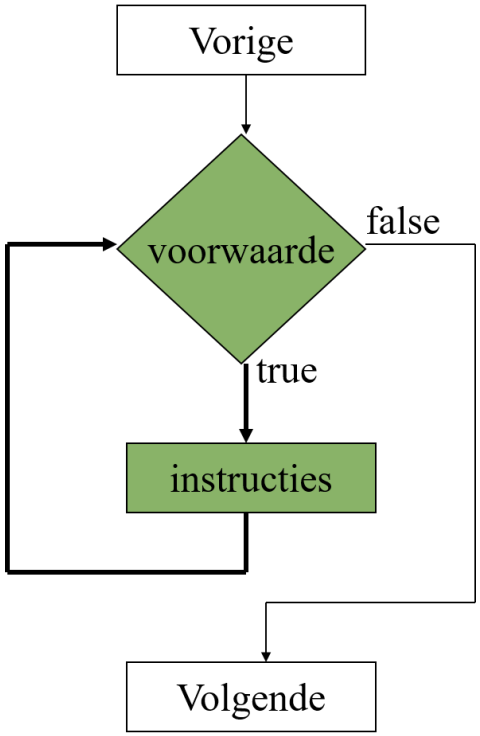
Algemene vorm van een do-while lus

Herhaal zolang een voorwaarde voldaan is. De voorwaarde wordt gecheckt pas nadat de body van de lus voor een eerste maal doorlopen werd. Deze body wordt dus minstens 1x uitgevoerd.

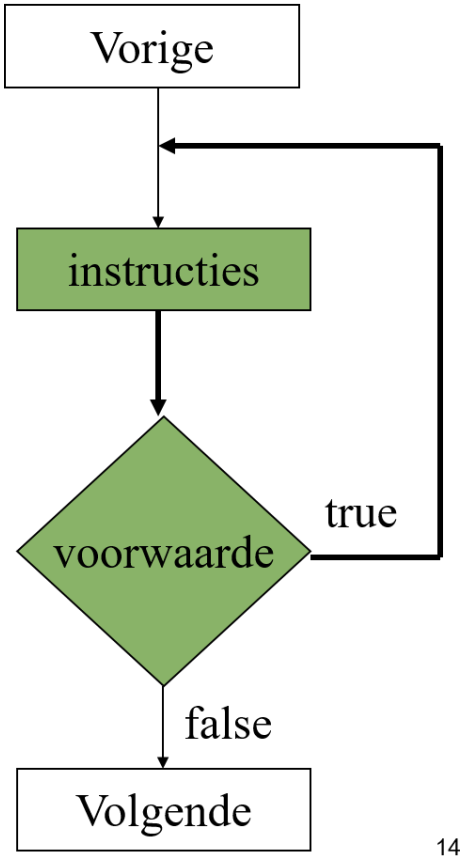
Algemene vorm van een `do/while`-lus:

```
do {
    ...
} while (booleaanse uitdrukking);
```

Flow diagrammen



While lus



Do-While lus

14

Blijf in de lus zolang het regent ...

```
boolean regen;

Scanner scan = new
    Scanner(System.in);
System.out.println("Regent
    het?");
regen =
    scan.nextBoolean();

while (regen) {
    System.out.println(
        "Regent het?");
    regen =
        scan.nextBoolean();
}
```

```
boolean regen;

Scanner scan = new
    Scanner(System.in);

do {
    System.out.println(
        "Regent het?");
    regen =
        scan.nextBoolean();
} while (regen)
```

Opletten met de stopconditie!

```
int count = 1;

while(count <= 25){
    System.out.println("count");
    count -= 1;
}
```

Via herhaling kan je input blijven vragen totdat deze correct is

```
Scanner input = new Scanner(System.in);
System.out.println("Geef een maand in
    (1..12)");
int maand = input.nextInt();

while(maand < 1 || maand > 12){
    System.out.println("Geef
       een maand in (1..12)");
     maand = input.nextInt();
}
```

Via herhaling kan je input blijven vragen totdat deze correct is

```
Scanner input = new Scanner(System.in);
int maand;

do {
    System.out.println("Geef
        een maand in (1..12)");
    maand = input.nextInt();
} while (maand < 1 || maand > 12);
```

Veel gemaakte (syntax)fouten

```
int getal;
Scanner scan = new Scanner(System.in);

while (getal <= 100) {
    getal = scan.nextInt(); //vraag getal
}
```

Veel gemaakte (syntax)fouten

```
int getal;
Scanner scan = new Scanner(System.in);

while (getal <= 100) {
    getal = scan.nextInt(); //vraag getal
}
```

Compiler error message :

variable might not have been initialized

Veel gemaakte (syntax)fouten

```
int getal = 123; // dummy waarde
Scanner scan = new Scanner(System.in);

while (getal <= 100) {
    getal = scan.nextInt(); //vraag getal
}
```

Veel gemaakte (syntax)fouten

```
int getal = 123; // dummy waarde
Scanner scan = new Scanner(System.in);

while (getal <= 100) {
    getal = scan.nextInt(); //vraag getal
}
```

error?

De compiler vindt van niet, maar ...
de lus zal niet starten, en dat was waarschijnlijk niet de bedoeling.
De dummy initialisatie van de variabele is niet goed gekozen!

Veel gemaakte (syntax)fouten

```
Scanner scan = new Scanner(System.in);

do {
    int getal = scan.nextInt(); //vraag getal
} while (getal <= 100);
```

Veel gemaakte (syntax)fouten

```
Scanner scan = new Scanner(System.in);

do {
    int getal = scan.nextInt(); //vraag getal
} while (getal <= 100);
```

Compiler error message :

Cannot find symbol
Symbol : variable getal

Oefening : Lengte van je tuin

Dit programma laat je toe de lengte van je tuin op te meten met een rolmeter van 5m lengte. Zolang je niet op het einde van je tuin bent gekomen, blijft het programma vragen of je de afsluiting van de tuin al bereikt hebt, en wordt er telkens 5m bijgeteld op de reeds gemeten afstand.

Oefening: Lengte van je tuin

Hoe de stopconditie formuleren ?

Wat is de output?

```
int lengteTuin = 0;
boolean bots = true;
    //true

while (!bots) {
    lengteTuin += 5;
}

System.out
      .println(lengteTuin);
```

Wat is de output?

```
int lengteTuin = 0;
boolean bots = false;
    //false

while (!bots) {
    lengteTuin += 5;
}

System.out
       .println(lengteTuin);
```

Oefening: Lengte van je tuin

```
public class LengteTuin {
 public static void main(String[] args) {
   Scanner scan = new Scanner(System.in);
   int rolmeter = 5; //in meter
   double lengteTuin = 0; //in meter

   System.out.println("Raak je de afsluiting al (j/n)
       ?");
   char antwoord = scan.next().charAt(0);

   while (antwoord != 'j' && antwoord != 'J') {
      lengteTuin += rolmeter;
      System.out.println("Raak je de afsluiting al
          (j/n) ?");
      antwoord = scan.next().charAt(0);
   }
```

Oefening: Lengte van je tuin

```
    System.out.println("LeesMeterAf (in cm):");
    lengteTuin += scan.nextInt() / 100.0;

    System.out.println("Je tuin is " +lengteTuin+ "
        meter lang");
  }
}
```

Wat gebeurt er als de gebruiker een foutieve input geeft, bijvoorbeeld een ?

Oefening: Lengte van je tuin

Oplossing 1

```
System.out.println("Raak je de afsluiting al (j/n)
    ?");
char antwoord = scan.next().charAt(0);

while (antwoord != 'j' && antwoord != 'J') {
    if (antwoord == 'n' || antwoord == 'N') {
    lengteTuin += rolmeter;
    }

    System.out.println("Raak je de afluiting al
        (j/n) ?");
    antwoord = scan.next().charAt(0);
}
```

Oefening: Lengte van je tuin

Oplossing 2

```
 char antwoord;

do {
   System.out.println("Raak je de afsluiting al
       (j/n) ?");
   antwoord = scan.next().charAt(0);

   if (antwoord == 'n' || antwoord == 'N') {
      lengteTuin += rolmeter;
   }
} while (antwoord != 'j' && antwoord != 'J');
```

Oefening: Lengte van je tuin

Oplossing 3

```
while (true) {
    System.out.println("Raak je de afsluiting al
        (j/n) ?");
    char antwoord = scan.next().charAt(0);

    if (antwoord == 'n' || antwoord == 'N') {
        lengteTuin += rolmeter;
    }
    else if (antwoord == 'j' || antwoord == 'J') {
        break;
    }
}
```

continue, break en return

```
public static void main(String[] args) {
   System.out.println("Geef getalwaarde: ");
   Scanner scan = new Scanner(System.in);
   int getal = scan.nextInt();

   for (int i = getal; i <
         Math.abs(getal*2); i++) {
      System.out.print(i);
      if (i < 0) {
         System.out.print("   ");
         continue;
      }
      else if (i > 0) {
         System.out.print("...");
         break;
      }
      else {
         System.out.print("---");
         return;
      }
   }
   System.out.println("!!!");
}
```

getal = 5
output : 5...!!!

getal = -3
output : -3 -2 -1 0—

getal = 0
output : !!!

break in geneste lussen

```
public static void main(String[] args) {
  for(int i=0; i< 3; i++){
    System.out.println("Buitenste loop : waarde i : " + i);
    System.out.print("Binnenste loop : waarde t : ");
    int t = 0;
    while(t<100){
        if (t==10) break;
        System.out.print( t + " ");
        t++;
    }
    System.out.println();
  }
  System.out.println("Einde van de loops");
}
```

break in geneste lussen

Printout naar het scherm

```
Buitenste loop : waarde i : 0
Binnenste loop : waarde t : 0 1 2 3 4 5 6 7 8 9
Buitenste loop : waarde i : 1
Binnenste loop : waarde t : 0 1 2 3 4 5 6 7 8 9
Buitenste loop : waarde i : 2
Binnenste loop : waarde t : 0 1 2 3 4 5 6 7 8 9
Einde van de loops
```

break in geneste lussen

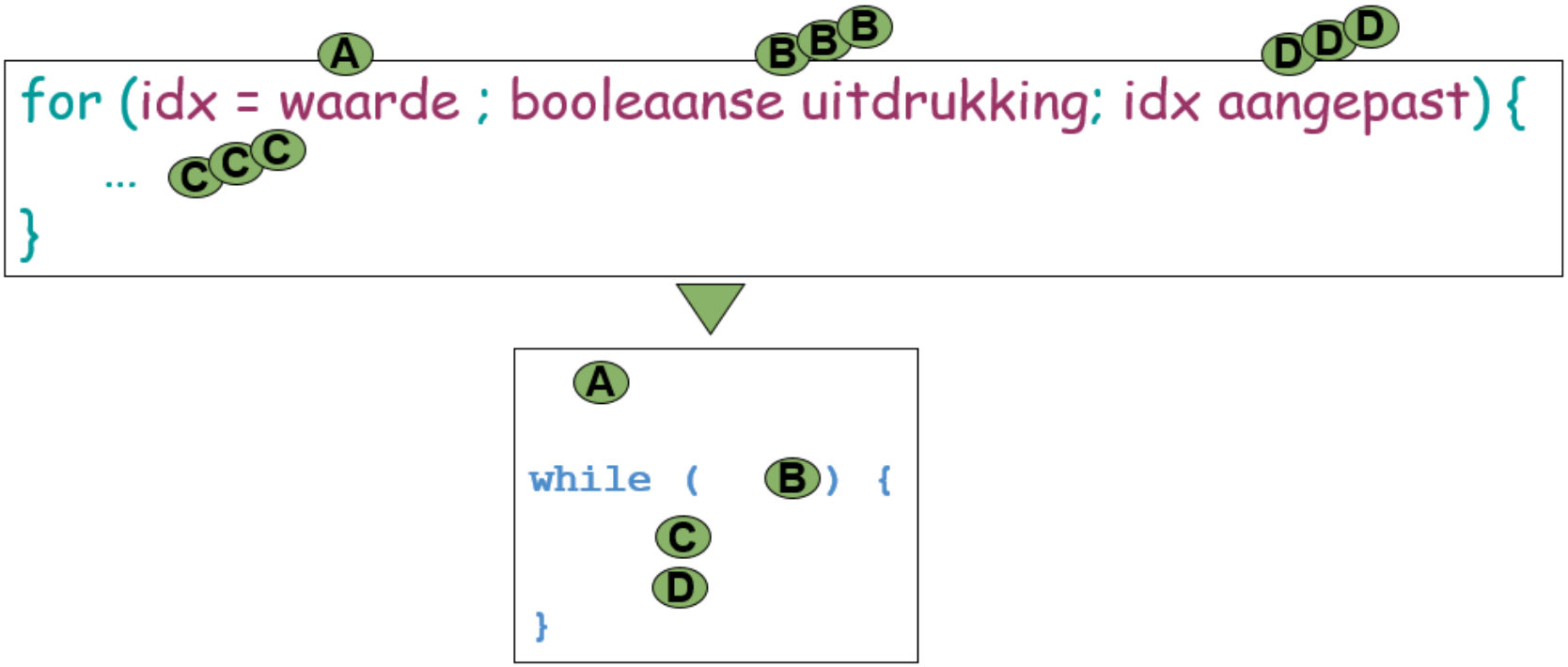
```
public static void main(String[] args) {
 one:
    for(int i=0; i< 3; i++){
        System.out.println("Buitenste loop : waarde i : ");
        System.out.print("Binnenste loop : waarde t : ");

        int t = 0;
 two:
    while(t<100){
        if (t==10) break one;
        System.out.print( t + " ");
        t++;
    }
    System.out.println();
 }
System.out.println("Einde van de loops");
}
```

break in geneste lussen

Printout naar het scherm

Buitenste loop : waarde i : 0
Binnenste loop : waarde t : 0 1 2 3 4 5 6 7 8 9 Einde van de loops

In elke for-lus herken je een while-lus



In elke for-lus herken je een while-lus

```
for (char c = 'A' ; c <= 'Z' ; c++) {
    System.out.println(c);
}
```

```
char c = 'A';

while (c <= 'Z') {
    System.out.println(c);
    c++;
}
```

Wanneer welke lus gebruiken?

- In principe maakt het niet uit, via elk lus-type kan je elke herhaling formuleren. Soms is de ene lus wel meer geschikt of eenvoudiger om te gebruiken dan de andere.
- Gebruik de **for-lus** wanneer je vooraf weet hoe vaak je zal moeten herhalen
- Gebruik de **do-while** wanneer je minstens 1x het codeblok moet uitvoeren
- Gebruik **while** wanneer je vooraf helemaal niet weet of en hoelang je zal moeten herhalen