

Module Leren Programmeren

OPERATOREN

C#-taal Basis - Overzicht

C# - Programmastructuur

C# - Syntax

C# - Variabelen

C# - Constants

C# - Data Types

C# - Data Type Conversie

C# - Operators

C# - Decision Making

- **C# - Loops**
- **C# - Encapsulation**
- **C# - Methods**
- **C# - Nullables**
- **C# - Arrays**
- **C# - Strings**
- **C# - Struct**
- **C# - Enums**

Inhoud LES 5 - Leren Programmeren

- **C# operatoren:**
 - **Rekenkundige operatoren**
 - **Logische operatoren**
 - **Vorrangsregels bij operatoren**
- **Conditie en decision making**

Operatoren bij numerieke datatypes

Operator	Beschrijving
+	Optellen
-	Aftrekken
*	Vermenigvuldigen
/	Delen
%	Modulus (Rest bij deling)
++	Incrementeren (1 bij optellen)
--	Decrementeren (1 van aftrekken)

Numerieke datatypes - Operatoren

- + - * / operator bij integer type (<https://www.learncs.org/>)

```
int a = 1;  
int b = 2;  
int c = a + b; // c is now equal to 3  
Console.WriteLine(c);
```

```
Console.WriteLine(3 - 5); // outputs -2  
Console.WriteLine(3 * 5); // outputs 15  
Console.WriteLine(3 / 5); // outputs 0!!!  
Console.WriteLine(3.0 / 5); // outputs 0.6!!!
```

Oefening

Vraag: wat geeft de volgende code als resultaat?

```
int getal1 = 1;  
int getal2 = 5;  
Console.WriteLine("de som is " + getal1 + getal2);  
Console.WriteLine("de som is " + (getal1 + getal2));
```

Voorbeelden numerieke operatoren

Addition (optelling)

```
int x = 5;  
int y = 2;  
int z = x + y;  
Console.WriteLine(z);
```

Subtraction (aftrekking)

```
Console.WriteLine(5-2);
```

Multiplication(vermenigvuldiging)

```
Console.WriteLine(5*2);
```

Division(deling): !opgelet bij delen van gehele getallen: altijd geheel getal als resultaat. Hoe los je dit op?

```
int x = 5;  
int y = 2;  
int z = x / y;  
Console.WriteLine(z);
```

Numerieke operatoren

Modulus (Rest bij deling)

```
int x = 5;  
int y = 2;  
int z = x % y;  
Console.WriteLine(z);
```

Increment(eren) (Unaire operator)

```
let x = 5;  
x++;  
Console.WriteLine(x);
```

Decrement(eren)(Unaire operator)

```
let x = 5;  
x--;  
Console.WriteLine(x);
```

(Assignment) toewijzings-Operatoren

worden meestal gebruikt bij numerieke variabelen

Operator	Voorbeeld	Zelfde als
=	x = y	x = y
+=	x += y	x = x + y
-=	x -= y	x = x - y
*=	x *= y	x = x * y
/=	x /= y	x = x / y
%=	x %= y	x = x % y

Oefeningen

Wat geven de volgende oefeningen als resultaat ?

Oefening 1

```
string textje = "Good ";  
textje += "Morning";  
Console.WriteLine(textje);
```

Oefening 2

```
int x = 10;  
x += 5;  
Console.WriteLine(x);
```

Oefening 3

```
int x = 10;  
x -= 5;  
Console.WriteLine(x);
```

Oefeningen – vervolg- wat is het resultaat?

Oefening 4

```
int x = 10;  
x *= 5;  
Console.WriteLine(x);
```

Oefening 5

```
int x = 10;  
x /= 5;  
Console.WriteLine(x);
```

Oefening 6

```
int x = 10;  
x %= 5;
```

Vorrangsregels van Operatoren

Vermenigvuldiging (*) en deling(/) hebben **voorrang** op optelling (+) en aftrekking(-)

Wat geven de volgende berekening als resultaat?

```
int x = 100 + 50 * 3;  
Console.WriteLine(x);
```

```
x = (100 + 50) * 3;  
Console.WriteLine(x);
```

```
int y = 100 + 50 - 3;  
Console.WriteLine(y);
```

Vergelijkingsoperatoren: geven true of false als resultaat (boolean)

Operator	Beschrijving	Voorbeeld
==	Is gelijk aan (waarde)	age == 18
!=	Is verschillend van	day != "Monday"
>	Is groter dan	salary > 9000
>=	Is groter dan of gelijk aan	salary >=9000
<	Is kleiner dan	age < 18
<=	Is kleiner dan of gelijk aan	age <= 18

VergelijkingsOperatoren voorbeelden

voorbeeld: `int x = 5;`

Operator	Beschrijving	Vergelijking	Resultaat
==	Is gelijk aan (waarde)	<code>x == 8</code>	false
		<code>x == 5</code>	true
!=	Is niet gelijk aan	<code>x != 8</code>	true
>	Is groter dan	<code>x > 8</code>	false
<	Is kleiner dan	<code>x < 8</code>	true
>=	Is groter dan of gelijk aan	<code>x >= 8</code>	false
<=	Is kleiner dan of gelijk aan	<code>x <= 8</code>	true

Voorrangsregels (boven= hoogste prioriteit)

Operator	Beschrijving	Voorbeeld
()	Expression grouping	(3 + 4)
++	Postfix Increment	i++
--	Postfix Decrement	i--
!	Logical not	!(x==y)
*	Multiplication	10 * 5
/	Division	10/5
%	Division Remainder	10 % 5
+	Addition	10 + 5
-	Subtraction	10-5
<	Less than	x < y
<=	Less than or equal	x <= y
>	Greater than	x > y
>=	Greater than or equal	x >= y
==	Equal	x == y
!=	Unequal	x != y
&&	Logical AND	x && y
	Logical OR	x y

Voorbeelden vergelijingsoperatoren

Oefening: Welke resultaten geven de volgende vergelijkingen?

```
int getal1 = 10;  
int getal2 = 5;
```

```
Console.WriteLine(getal1 == getal2);  
Console.WriteLine(getal1 != getal2);  
Console.WriteLine(getal1 > getal2);  
Console.WriteLine(getal1 >= getal2);  
Console.WriteLine(getal1 < getal2);  
Console.WriteLine(getal1 <= getal2);
```

Booleans en Booleaanse operatoren

Booleaanse waarden (booleans) zijn ofwel **true** ofwel **false**
voorbeelden van variabele declaratie voor booleans:

bool isOud = **false**;

bool isSchuldig = **true**;

Operatoren op booleans

& & (AND)

| | (OR)

! (NOT)

== (gelijkheid)

!= (niet gelijk aan)

Conditionele operatoren

Operator	Description	Example
==	equal to	if (day == "Monday")
>	greater than	if (salary > 9000)
<	less than	if (age < 18)

Voorbeeld van gebruik: bij testen van een conditie

```
int age = 16;  
if (age < 18)  
{  
    Console.WriteLine("Te jong!");  
}
```

Condities in C#

Syntax conditie:

```
if (conditie) {  
    codeblok wordt uitgevoerd als conditie true is  
}
```

Voorbeeld van gebruik:

```
int hour = 16;  
if (hour < 18)  
{  
    Console.WriteLine("Good day");  
}
```

Oefening:

vraag hoe laat het is (geheel getal tussen 0 en 23) en lees het opgegeven uur in.

Tip: `int uur = int.Parse(Console.ReadLine());`

Schrijf "Goede morgen" naar de console indien het getal kleiner is dan 12



Boolean en condities

Demo

Boolean en condities

Oefeningen

Oefeningen C#:

https://www.w3schools.com/cs/exercise.asp?filename=exercise_booleans1

https://www.w3schools.com/cs/exercise.asp?filename=exercise_booleans2

https://www.w3schools.com/cs/exercise.asp?filename=exercise_conditions1

Oefening condities - 1

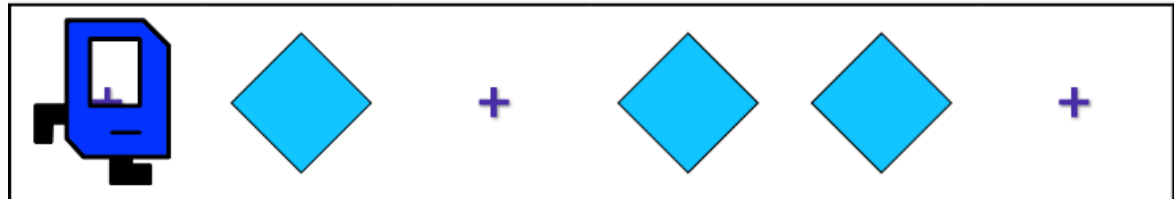
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 1: intro-12: Schrijf code die alle beeper opneemt

Gebruik de volgende conditie om de beepers op de nemen:

```
if (beepersPresent())  
{  
    pickBeeper();  
}
```



Oefening condities – 2

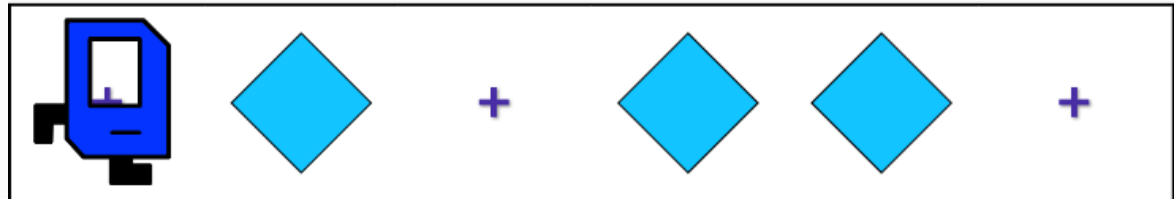
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 2: intro-12:

Wat gebeurt er indien je de condities uit Oefening 1 vervangt door de volgende code:

```
if(beepersPresent())  
{  
    pickBeeper();  
}  
else  
{  
    putBeeper();  
}
```



Oefening condities – 3

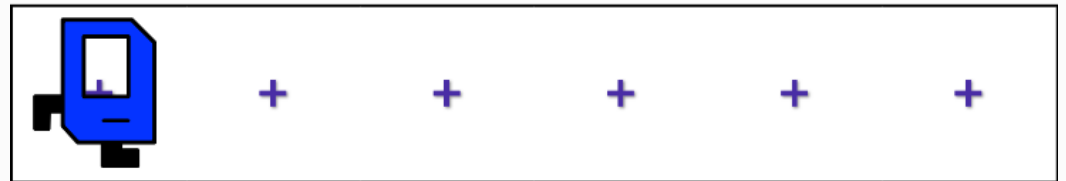
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 3: intro-01:

Schrijf de onderstaande code. Wat krijg je als resultaat?

```
function main() {  
    while (frontIsClear())  
    {  
        move();  
    }  
}
```



Oefening condities – 4

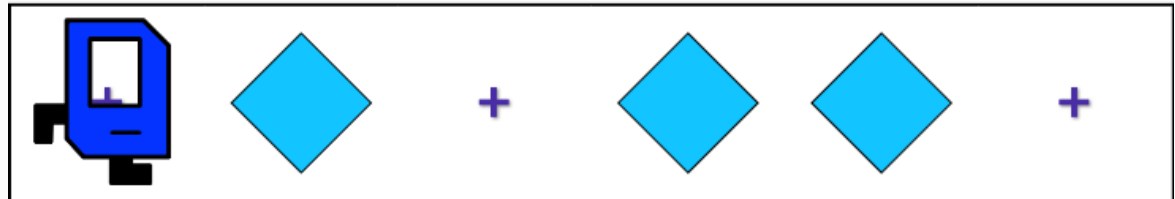
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 4: intro-12:

Vervang de code door de volgende code. Wat krijg je als resultaat?

```
function main() {  
  
    while(frontIsClear())  
    {  
        if(beepersPresent())  
        {  
            pickBeeper();  
        }  
        else  
        {  
            putBeeper();  
        }  
        move();  
    }  
}
```



Oefening condities – 5

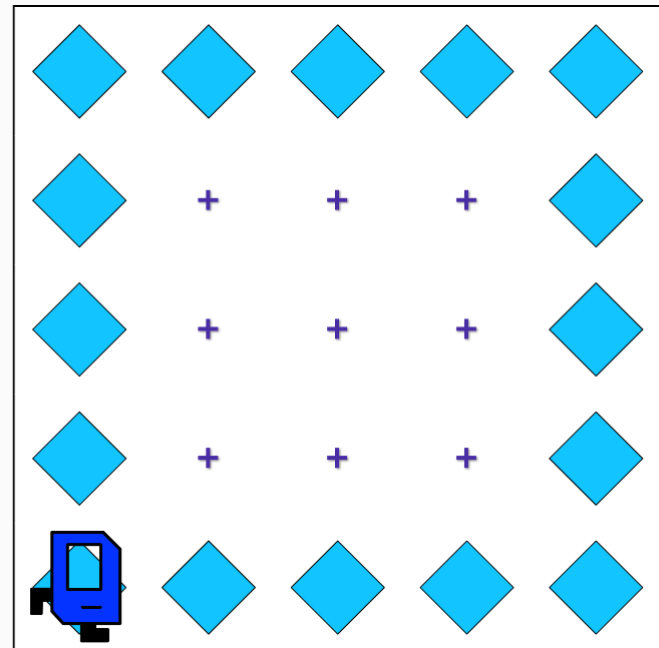
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 5: intro-06:

Schrijf de volgende code. Wat krijg je als resultaat? Pas de code aan zodat alle beepers worden opgenomen

```
function main() {  
  repeat(4)  
  {  
    while(frontIsClear())  
    {  
      move();  
    }  
    turnLeft();  
  }  
}
```



Oefening condities – 6

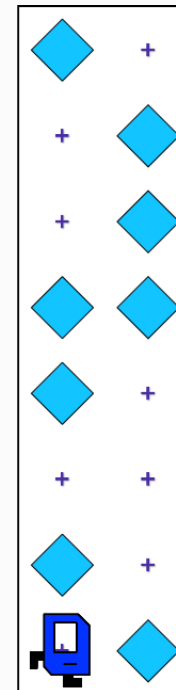
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 6: intro-13:

Schrijf de volgende code. Wat krijg je als resultaat? Pas de code aan zodat alle beepers worden opgenomen

```
function main() {  
    turnLeft();  
    while(frontIsClear())  
    {  
        move();  
    }  
    turnRight();  
    move();  
    turnRight();  
    while(frontIsClear())  
    {  
        move();  
    }  
}
```



Oefening condities – 7

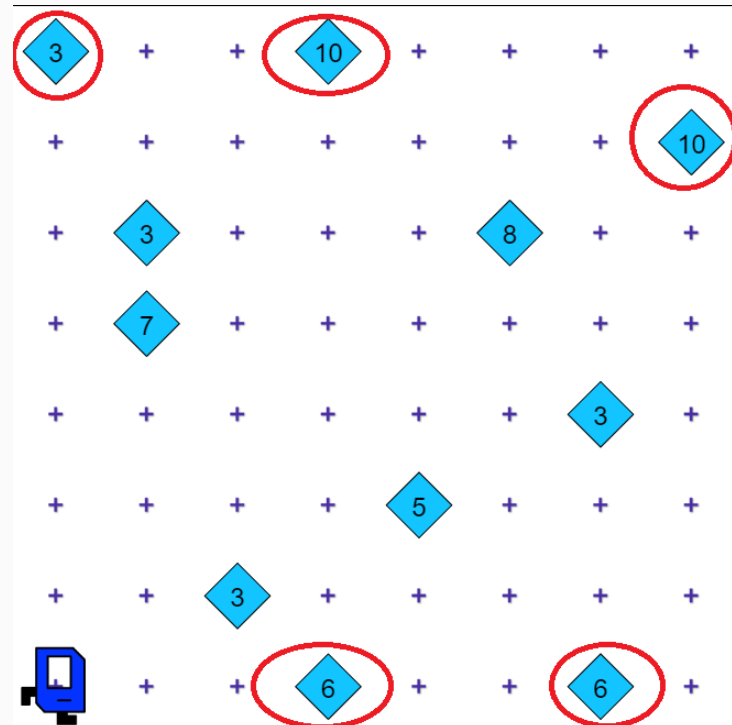
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 7: beepers-8x8:

Schrijf de volgende code. Wat krijg je als resultaat? Pas de code aan zodat alle beepers langs de randen worden opgenomen

```
function main() {  
    repeat(4)  
    {  
        while(frontIsClear())  
        {  
            move();  
            if(beepersPresent())  
            {  
                pickBeeper();  
            }  
        }  
        turnLeft();  
    }  
}
```



Oefening condities – 8

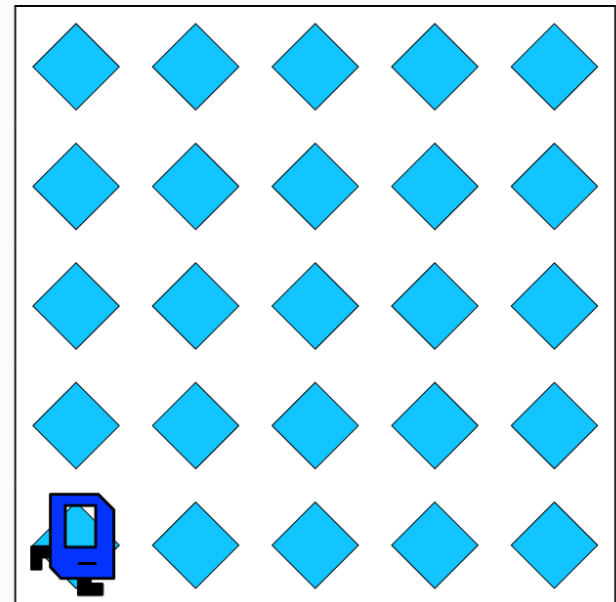
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 8: intro-08:

Schrijf de volgende code. Wat krijg je als resultaat? Pas de code aan zodat alle beepers worden opgenomen

```
function main() {  
  repeat(2)  
  {  
    repeat(4)  
    {  
      move();  
    }  
    turnLeft();  
    move();  
    turnLeft();  
    repeat(4)  
    {  
      move();  
    }  
    turnRight();  
    move();  
    turnRight();  
  }  
  repeat(4)  
  {  
    move();  
  }  
}
```



Oefening condities – 9

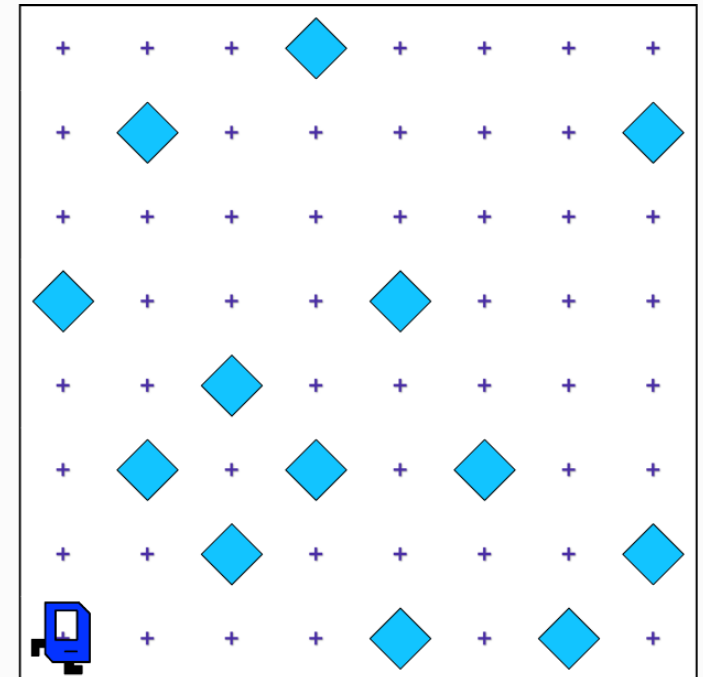
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 9: intro-14:

Schrijf de volgende code. Wat krijg je als resultaat? Pas de code aan zodat alle beepers worden opgenomen

```
function main() {  
  repeat(4)  
  {  
    turnLeft();  
    repeat(7)  
    {  
      move();  
    }  
    turnRight();  
    move();  
    turnRight();  
    repeat(7)  
    {  
      move();  
    }  
    turnLeft();  
    if (frontIsClear())  
    {  
      move();  
    }  
  }  
}
```



Oefening condities – 10

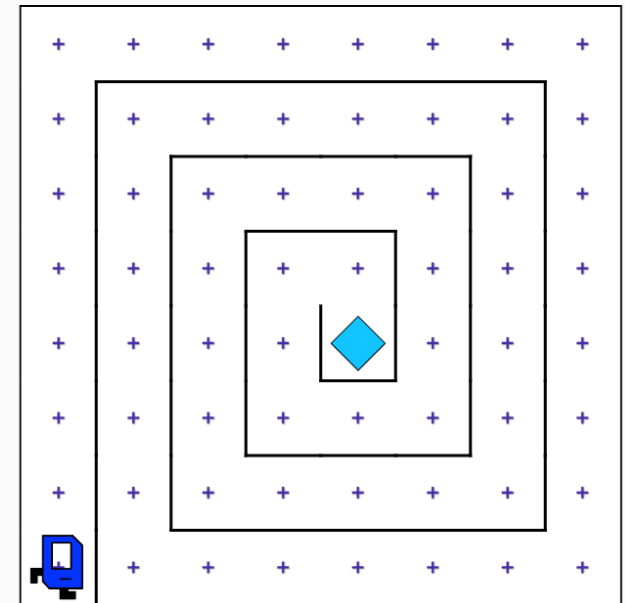
Oefeningen condities:

Oefeningen Karel/Karen op <https://www.kareltherobot.ch/karel.html>

Oefening 10: intro-15:

Schrijf de volgende code. Wat krijg je als resultaat? Pas de code aan zodat de beeper in het midden wordt opgenomen

```
function main() {  
    turnLeft();  
    while (noBeepersPresent())  
    {  
        //Hier code toevoegen  
        turnRight();  
    }  
    pickBeeper();  
}
```



Oefeningen online C#

Oefeningen die je reeds kan maken:

C# variabelen:

https://www.w3schools.com/cs/exercise.asp?filename=exercise_variables1

Exercise 1 -> 4

C# Data Types:

https://www.w3schools.com/cs/exercise.asp?filename=exercise_data_types1

Exercise 1 -> 4

C# User Input:

https://www.w3schools.com/cs/exercise.asp?filename=exercise_user_input1

Exercise 1

Oefeningen online C# - vervolg

Oefeningen die je reeds kan maken:

C# operators:

https://www.w3schools.com/cs/exercise.asp?filename=exercise_operators4

Exercise 1 -> 4

https://www.w3schools.com/cs/exercise.asp?filename=exercise_strings1

Exercise 1 and 2

https://www.w3schools.com/cs/exercise.asp?filename=exercise_booleans1

Exercise 1->2

Referenties

- <https://www.w3schools.com/cs>
- <https://www.learncs.org/>
- <https://dotnetfiddle.net/>