

```
int GrootsteGemeneDeler(int getal1, int getal2)
{
    while (getal2 != 0)
    {
        int rest = getal1 % getal2;
        getal1 = getal2;
        getal2 = rest;
    }

    return getal1;
}

static void Main(string[] args)
{
    int getal1 = Int32.Parse(Console.ReadLine());
    int getal2 = Int32.Parse(Console.ReadLine());
    int ggd = GrootsteGemeneDeler(getal1, getal2);
    Console.WriteLine("{0} is de GGD van {1} en {2}.",
        ggd, getal1, getal2);
    Console.ReadKey();
}
```

CO
- ORIGIN
'warn'.

monitor

thing 2
duties 3
picture

Programmeren 1 (C#)

Gerwin van Dijken (gerwin.vandijken@inholland.nl)

Programmeren jaar 1

- Programmeren jaar 1: Visual Studio / C#
- Elke periode een (praktijk) toets + verplichte opdrachten
- Thema 1.1: **Programmeren 1** (3 EC)
 - Programmeren 1 (100%)
 - Opdrachten Programmeren 1 (0%) ✓
(imperatief programmeren, Console/Windows applicaties)
- Thema 1.2: **Programmeren 2** (4 EC)
 - Programmeren 2 (100%)
 - Opdrachten Programmeren 2 (0%) ✓
(imperatief programmeren, Console/Windows applicaties)

Programmeren jaar 1

- Thema 1.3: **Programmeren 3** (3 EC)

- Programmeren 3 (100%)
 - Opdrachten Programmeren 3 (0%) ✓
- (object-georiënteerd programmeren)*

- Thema 1.4: **Design Patterns** (3 EC)

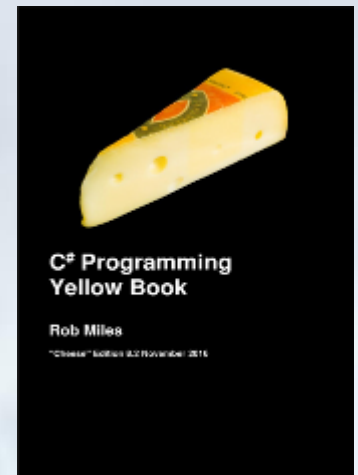
- Design Patterns (100%)
 - Opdrachten Design Patterns (0%) ✓
- (flexibiliteit, toepassen van algemene software oplossingen)*
- Project Applicatiebouw (5 EC's)

Programma periode 1.1 (Programmeren 1)

01 (wk-36)	Inleiding / Visual Studio 2017/2019
02 (wk-37)	Sequentie
03 (wk-38)	Selectie
04 (wk-39)	Iteratie
05 (wk-40)	Array's
06 (wk-41)	Methoden
07 (wk-42)	Herhaling / oefententamen
08 (wk-43)	<i>roostervrije week</i>
09 (wk-44)	praktijktentamen (<i>computer opdrachten</i>)
10 (wk-45)	-

Materiaal

- Moodle course:
1920 Inf1.1S Webdesign (*zelf enrollen*)
 - Powerpoint slides
 - wekelijkse opdrachten
- Boek (*aanvullend*):
'The C# Programming Yellow Book'
(*"Cheese" Edition 8.2, November 2016*)
 - *download link op Moodle*



Toetsing

- Programmeren 1 (1918IN113A)
 - praktijktentamen (computer)
 - Visual Studio 2013/2015/2017/2019
 - maken van (Console/Windows) applicaties
- Opdrachten Programmeren 1 (1918IN113B)
 - verplichte wekelijkse opdrachten
 - vanaf week 2 (5x), afvinken steeds 1 week later
 - week 1: introductie

Programmeren 1		INF	1918IN113Z	0 van 3	
-	- Programmeren 1	INF	1918IN113A	100	GRD
-	- Opdrachten Programmeren 1	INF	1918IN113B	0	SUS

Colleges

- Hoorcollege

- theorie, probleem analyse, oefeningen, demo's, vragen

- Praktijkcollege

- verplichte wekelijkse opdrachten
- Microsoft Visual Studio 2013/2015/2017/2019
- één solution per week met daarin alle opdrachten

week 1

INTRODUCTIE PROGRAMMEREN

Programmeer paradigma's

- Er zijn verschillende concepten van programmeren

- **Imperatief programmeren** (o.a. C, Pascal)

uitvoeren van commando's, toestand

```
printf("Hello World!\n");
```

- Object-georiënteerd programmeren (o.a. **C#**, Java)

werken met objecten

```
class Fiets : Voertuig { ... }
```

- Functioneel programmeren (o.a. Haskell, ML)

definieren / uitvoeren van functies

```
> list1 = [1, 2, 3, 4, 5]  
> filter odd list1
```

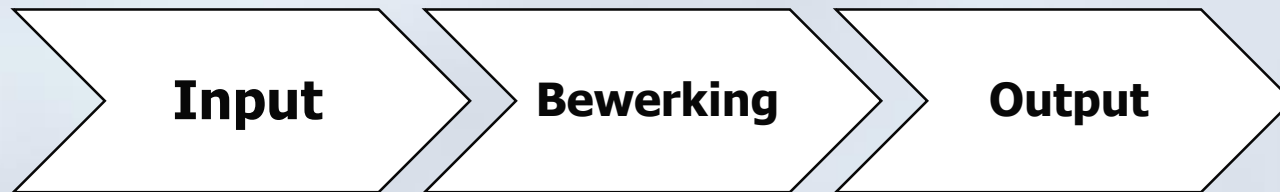
- Logisch programmeren (o.a. Prolog)

```
ouder_van(julia, augustus)  
vrouw(julia)  
moeder_van(X,Y) :- ouder_van(X, Y), vrouw(X)
```

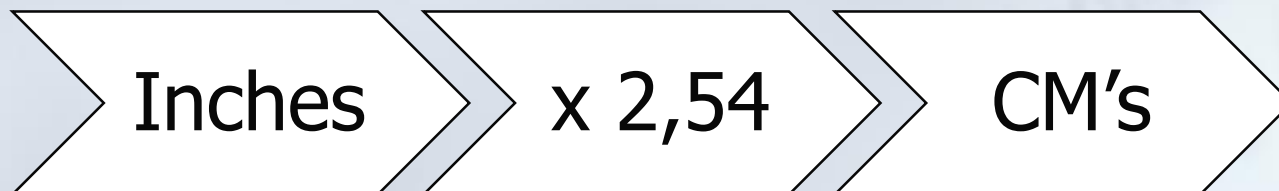
Software ontwikkeling

Basis principe van een (imperatief) software programma

Generiek



De oplossing van een probleem (voorbeeld)

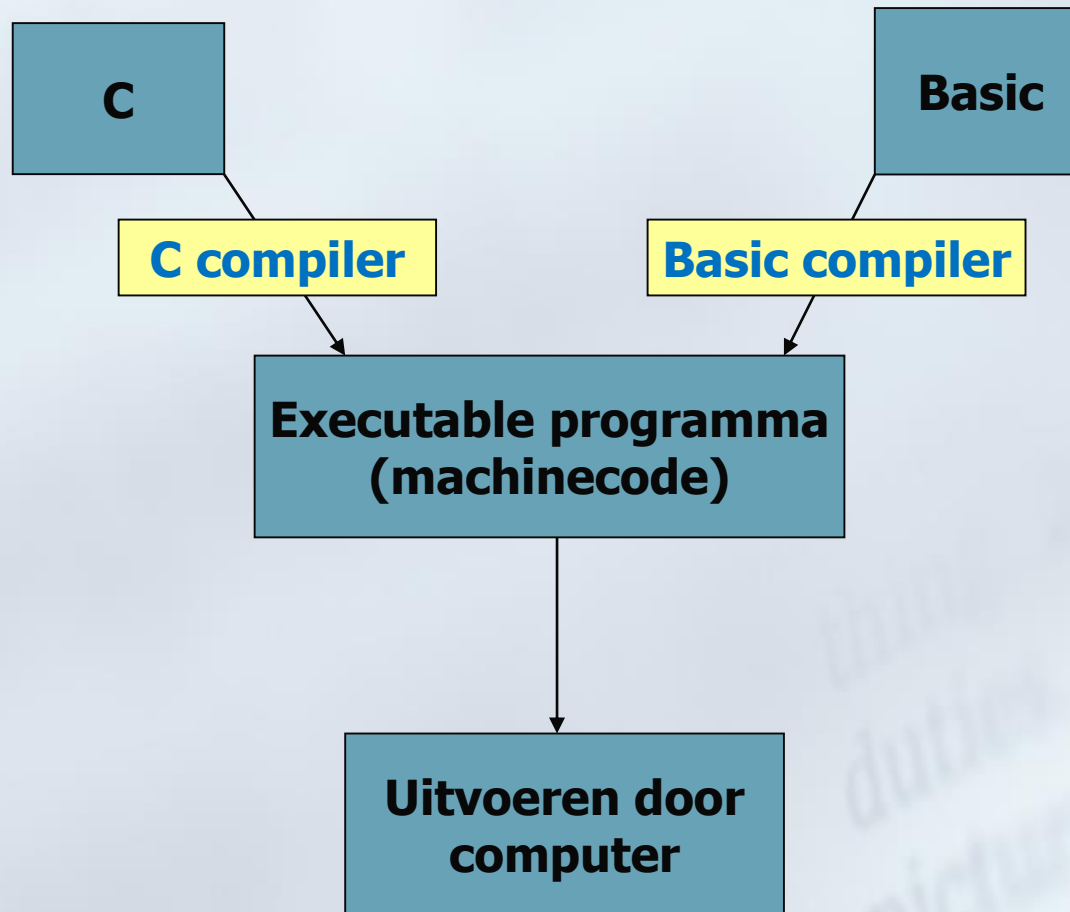


Programmeertalen

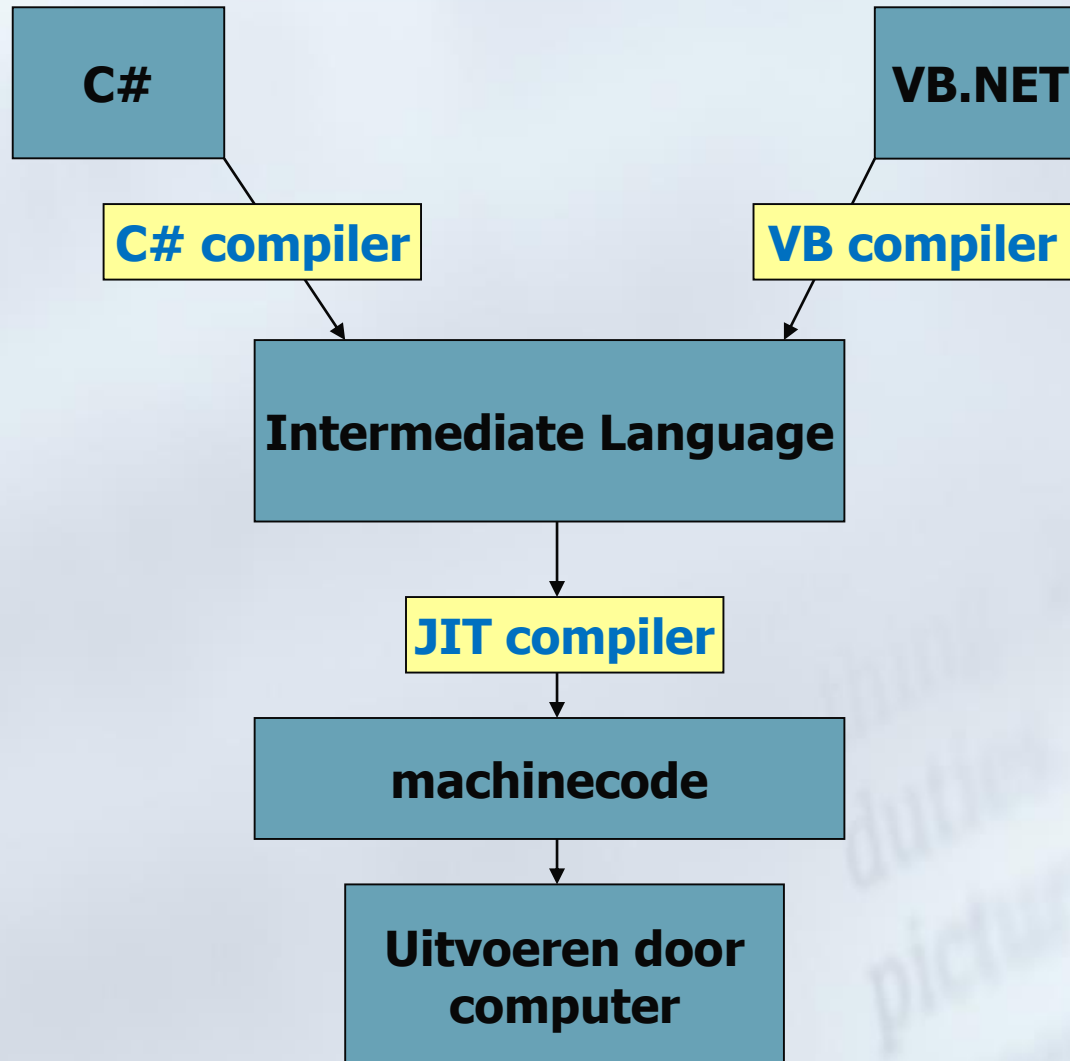
- Vroeger machinecode / mnemonic codes
bv "MOV AX, BX" / "DEC AX"
geen vertaling nodig, zeer complex
- Compilers om leesbare tekst om te zetten naar machinecode
machinecode wordt door CPU verwerkt
- BASIC, C, C++, JAVA, C#
- Intermediate language



Algemeen programmeren



Programmeren in .NET



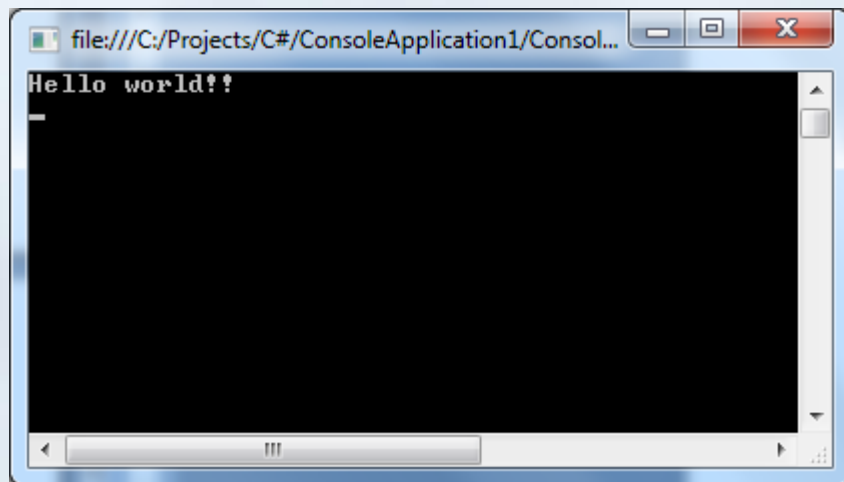
Ontwikkelomgevingen

- Voor ontwikkelen van software: IDE
(Integrated Development Environment)
- IDE-applicatie bevat meerdere functies
(tekst highlighting, intellisense, debugging, versiebeheer, grafische ontwerper, ...)
- Java → Eclipse, IntelliJ
- C# → Visual Studio 2013/2015/2017/2019

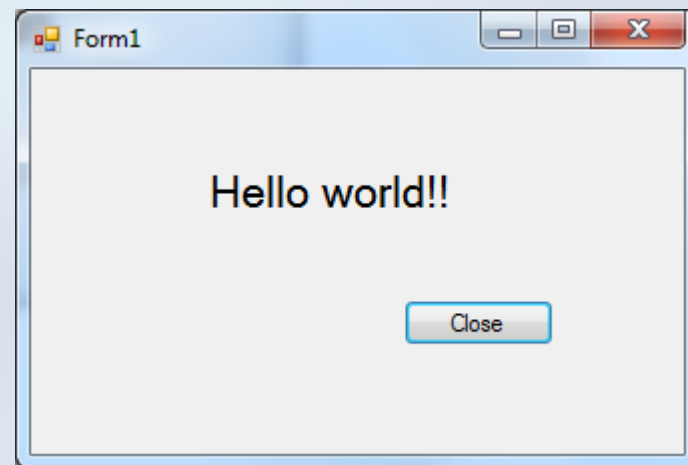
Type programma's

- 1^{ste} jaar
 - Console applicaties
 - Windows (Forms) applicaties / WPF applicaties
- 2^{de} jaar
 - Webapplicaties (PHP, ModelViewController)
 - Java

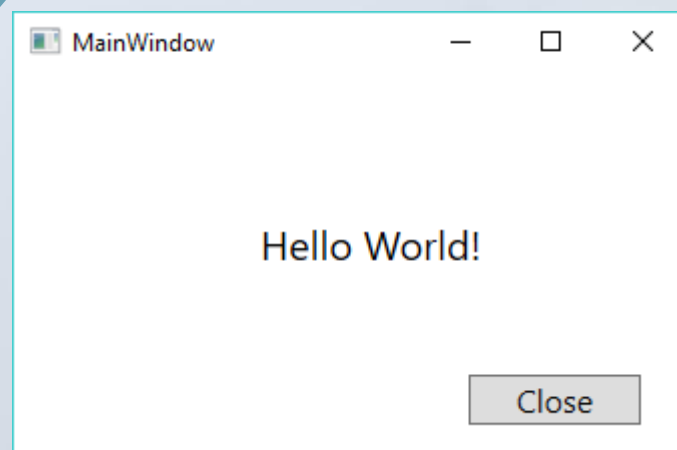
Console vs Windows applicaties



Console Application



Windows Forms Application



Windows Presentation Foundation (WPF)

- We leren de basis van programmeren met Console en Windows applicaties (*thema 1.1*)

Visual Studio - demo



Visual Studio

(aanmaken van een Console / Windows Forms applicatie)

Variabelen

- Een variabele bevat een waarde die kan veranderen
- Een naam van een variabele:
 - begint met een letter of _
 - daarna letter, cijfer, of _
 - geen reserveerde namen (if, for, while, ...)
 - afspraak: beginnend met een kleine letter

```
int getal, _antwoord, a3, 3tal, false;
```

Datatypen

- Een variabele is van een bepaald type
- De mogelijke waarden van een variabele is afhankelijk van het datatype
- Primitieve datatypen (build in):
 - byte / short / **int** / long
 - **float** / **double** / decimal
 - **char** / **string**
 - **bool** (true / false)

```
int getal, _antwoord;  
string naam;  
char keuze;  
bool gevonden;  
float lengte;
```

Visual Studio - demo



(declaratie van variabelen)

int, double, boolean

De belangrijkste (meest gebruikte) drie zijn:

- int

gehele getallen zoals:

0, -25, 1000000

- double

gebroken getallen zoals:

3.14, -25.555555

- boolean

waar of niet-waar: true, false

Operatoren voor int

+	optellen	3 + 4	(7)
-	afrekken	5 - 2	(3)
*	vermenigvuldigen	4 * 5	(20)
/	gehele deling (<i>quotiënt</i>)	10 / 3	(3)
%	modulo (<i>restwaarde</i>)	10 % 3	(1)

```
int hoogte1, hoogte2, totaleHoogte;  
  
hoogte1 = 75;  
hoogte2 = 200;  
totaleHoogte = hoogte2 + hoogte1;           // 275  
totaleHoogte = hoogte2 / hoogte1;           // 2  
totaleHoogte = hoogte2 % hoogte1;           // 50
```

Operatoren voor double

+	optellen	3.1 + 4.0
-	afrekken	5 - 2.1
*	vermenigvuldigen	4.0 * 5.1
/	<i>gewone</i> deling	10 / 3

```
double p, q, r, s, t;  
p = 3.1 + 4.0; // 7.1  
q = 5 - 2.1;   // 2.9  
r = 4.0 * 5.1; // 20.4  
s = 10 / 3;    // 3 (!)  
t = 10.0 / 3;  // 3.33333333
```

Visual Studio - demo



Visual Studio

(operatoren toepassen op int/double variabelen)

Datatype bool / Boolean

- Het primitieve type boolean kent slechts twee verschillende waarden: true en false
- Een boolean krijg je meestal als resultaat van het vergelijken van twee waarden
- Het vergelijken gaat met relationele operatoren

Relationele operatoren

>	groter dan
<	kleiner dan
>=	groter dan of gelijk aan
<=	kleiner dan of gelijk aan
==	gelijk aan
!=	ongelijk aan

Een boolean voorbeeld (1)

```
int maxStudenten = 30;  
int aantalStudenten = 24;  
bool result;  
  
result = (aantalStudenten < maxStudenten);  
result = (aantalStudenten <= maxStudenten);  
result = (aantalStudenten != maxStudenten);
```

```
int maxStudenten = 30;  
int aantalStudenten = 24;  
bool result;  
  
result = (aantalStudenten < maxStudenten);      // true  
result = (aantalStudenten <= maxStudenten);      // true  
result = (aantalStudenten != maxStudenten);      // true
```

Een boolean voorbeeld (2)

```
int maxStudenten = 30;
int aantalStudenten = 24;
bool result;

result = (aantalStudenten == maxStudenten);
result = (aantalStudenten >= maxStudenten);
result = (aantalStudenten > maxStudenten);
```

```
int maxStudenten = 30;
int aantalStudenten = 24;
bool result;

result = (aantalStudenten == maxStudenten);    // false
result = (aantalStudenten >= maxStudenten);    // false
result = (aantalStudenten > maxStudenten);     // false
```

Constanten

- Een constante bevat een waarde die niet verandert
- Een constante wordt op dezelfde manier gedeclareerd als een variabele, alleen met 'const' ervoor
 - een constante moet direct een waarde krijgen
 - afspraak: alles met een hoofdletter

```
const int MIN_LEEFTIJD = 6;  
const int MAX_LEEFTIJD = 90;  
const float LENGTH;
```

Visual Studio - demo



(een constante definiëren)

Operator prioriteit

- Prioriteit van hoog naar laag:
 - + (unair) en - (unair)
 - * / %
 - + (binair) en - (binair)
- Gelijke prioriteit → (evalueer) van links naar rechts
- Met haakjes verander je de prioriteit

```
int x1, x2, x3, x4, x5;  
  
x1 = 3 + 4 * 2;      // 11  
x2 = (3 + 4) * 2;    // 14  
x3 = 9 / 4;          // 2  
x4 = 9 % 4;           // 1  
x5 = 40 / 5 * 2;     // 16
```

Extra rekenkundige operatoren

++	unair (1)	verhogen met 1
--		verlagen met 1
+=	binair (2)	optellen en toekennen
-=		afrekken en toekennen
*=		vermenigvuldigen en toekennen
/=		delen en toekennen
%=		module/restwaarde en toekennen

```
int x1 = 1, x2 = 2, x3 = 3;  
x1++;           // 2  
x2--;           // 1  
x3 += 4;        // 7
```


char en string

- char kan '1' karakter bevatten
- string kan "meerdere" karakters bevatten

```
char letter = 'g';  
char letter2 = "x";  
string woord = "meerdere_karakters";  
string woord2 = 'blablabla';
```

Too many characters in character literal

Huiswerk (voor volgende week)

- Lezen:

- Yellow Book (H1: Computers and Programs, page 2..13)