Informatik I für D-ITET Musterlösung 6 WS 04/05

URL: http://www.ti.inf.ethz.ch/ew/courses/inf1-ITET/

Aufgabe 1

a) Programm: squares.C

```
#include <IFET/turtle>

void square(unsigned int i)
{
  for (unsigned int j = 0; j < 4; ++j) {
    ifet::forward(i);
    ifet::right(90);
  }
}

int main()
{
  for (unsigned int j = 1; j <= 2; ++j)
    for (unsigned int i = 0; i < 8; ++i) {
      square(j);
      ifet::right(45);
    }
  return 0;
}
```

b) Programm: spiral.C

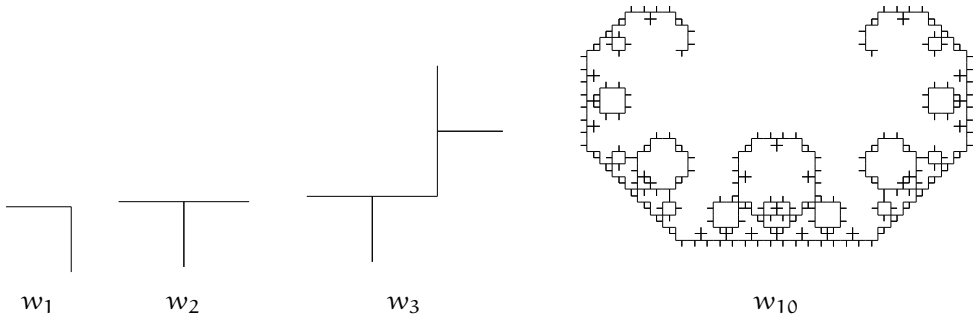
```
#include <IFET/turtle>

int main()
{
  for (unsigned int i = 1; i < 17; ++i)
    for (unsigned int j = 0; j < 2; ++j) {
      ifet::forward(i);
      ifet::right(90);
    }
  return 0;
}
```

c) Programm: lind.C

```
// Programm: lind.C
// Stellt die Woerter des Lindenmayer-Systems
// ({X,+},{X->X+X+, +->+},X) mit dem Winkel 60 Grad
// grafisch dar.

#include <iostream>
#include <IFET/turtle>

void x(unsigned int i)
{
  if (i == 0)
```

```
    ifet::forward();
  else {
    x(i-1);
    ifet::right(90);
    x(i-1);
    ifet::right(90);
  }
}

int main()
{
  std::cout << "Wieviele Iterationen? ";
  unsigned int k;
  std::cin >> k;
  x(k);
  return 0;
}
```

Ausgabe:



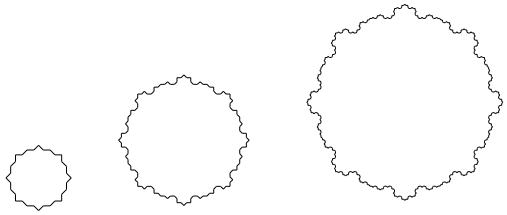
Aufgabe 2

a) Programm: lind-ko.C

```
// Programm: lind-ko.C
// Zeichnet die Koch-Kurve.

#include <iostream>
#include <IFET/turtle>

void X(unsigned int n)
{
  if (n == 0)
    ifet::forward();
  else {
    X(n-1);
    ifet::left(45);
    X(n-1);
    ifet::right(90);
    X(n-1);
    ifet::left(45);
    X(n-1);
  }
}
```

```
int main()
{
  std::cout << "Wieviele Iterationen? ";
  unsigned int k;
  std::cin >> k;
  for (unsigned int i = 0; i < 8; ++i) {
    X(k);
    ifet::right(45);
  }
  return 0;
}
```

Ausgabe:



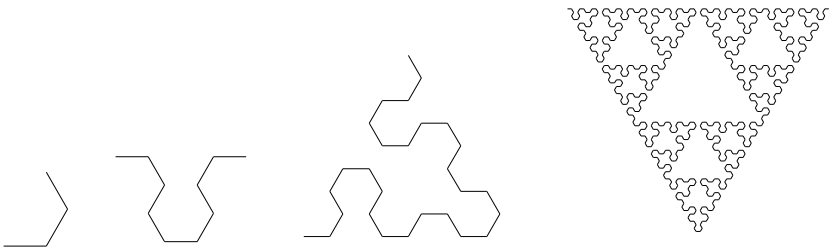
w_1 w_2 w_3

b) Programm: lind-st.C

```
// Programm: lind-st.C
// Zeichnet das Sierpinski-Dreieck.

#include <iostream>
#include <IFET/turtle>

void y(unsigned int r);

void x(unsigned int r)
{
  if (r == 0)
    ifet::forward();
  else {
    y(r - 1);
    ifet::right(60);
    x(r - 1);
    ifet::right(60);
    y(r - 1);
  }
}

void y(unsigned int r)
{
  if (r == 0)
    ifet::forward();
  else {
    x(r - 1);
    ifet::left(60);
    y(r - 1);
    ifet::left(60);
    x(r - 1);
  }
```

```
}

int main()
{
  std::cout << "Wieviele Iterationen? ";
  unsigned int k;
  std::cin >> k;
  y(k);
}
```

Ausgabe:



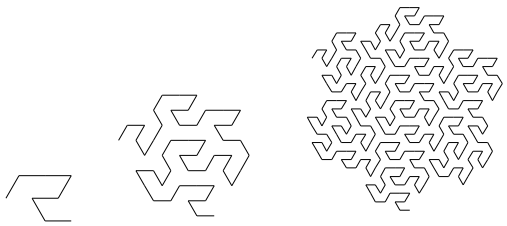
w_1 w_2 w_3 w_6

c) Programm: lind-go.C

```
// Programm: lind-go.C
// Zeichnet die Gosper-Kurve.

#include <iostream>
#include <IFET/turtle>

void y(unsigned int r);

void x(unsigned int r)
{
  if (r == 0)
    ifet::forward();
  else {
    x(r - 1);
    ifet::right(60);
    y(r - 1);
    ifet::right(120);
    y(r - 1);
    ifet::left(60);
    x(r - 1);
    ifet::left(120);
    x(r - 1);
    x(r - 1);
    ifet::left(60);
    y(r - 1);
    ifet::right(60);
  }
}

void y(unsigned int r)
{
  if (r == 0)
    ifet::forward();
  else {
```

```
    ifet::left(60);
    x(r - 1);
    ifet::right(60);
    y(r - 1);
    y(r - 1);
    ifet::right(120);
    y(r - 1);
    ifet::right(60);
    x(r - 1);
    ifet::left(120);
    x(r - 1);
    ifet::left(60);
    y(r - 1);
  }
}

int main()
{
  std::cout << "Wieviele Iterationen? ";
  unsigned int k;
  std::cin >> k;
  y(k);
}
```

Ausgabe:



w_1 w_2 w_3

Aufgabe 3

a) $\Sigma = \{X, +, -\}$, $P = \{X \mapsto X\text{-}X+X+X\text{-}X,\ + \mapsto +,\ - \mapsto -\}$, $s = X\text{-}X\text{-}X\text{-}X\text{-}$ und $\alpha = 90°$.

b) $\Sigma = \{X, +, -\}$, $P = \{X \mapsto XX+X+X+X+X+X\text{-}X,\ + \mapsto +,\ - \mapsto -\}$, $s = X\text{-}X\text{-}X\text{-}X\text{-}$ und $\alpha = 90°$.

c) $\Sigma = \{X, +, -\}$, $P = \{X \mapsto XX\text{-}X\text{-}X\text{-}X\text{-}XX,\ + \mapsto +,\ - \mapsto -\}$, $s = X\text{-}X\text{-}X\text{-}X\text{-}$ und $\alpha = 90°$.