Dr. Andreas Borchert
Michael Wiedemann
Abteilung Angewandte Informationsverarbeitung
27. April 2004
Blatt 1

Allgemeine Informatik II (SS 2004)

Abgabetermin: 6. Mai 2004

1 Pflanzen malen – 10 Punkte

Zu Beginn des neuen Semesters lassen wir es dementsprechend langsam angehen, und tauchen vorsichtig in die Welt der Rekursionen ein. Um das ganze ein ganz klein wenig spannender zu gestalten, bedienen wir uns der Möglichkeit unter UNIX, geometrische Objekte zeichnen zu können. Hierzu zunächst eine kurze Einführung.

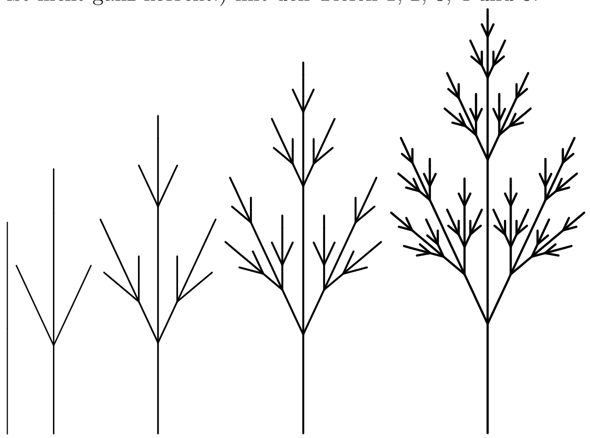
Wir betrachten zwei Module, mit deren Hilfe wir etwas zeichnen und entweder in einem Fenster oder in eine Datei ausgeben können.

Das Modul **Plot5Streams bzw. Plotters**: Mit dem Modul *Plotters* kann unter Verwendung eines zweidimensionalen Koordinatensystem direkt gezeichnet werden.
Das Modul **TurtleGraphics**: Mit diesem Modul kann kann eine Zeichnung in das System von Plotters uebertragen werden. Hierbei kann man sich das ganze folgendermaßen vorstellen: Maßgebend sind die aktuelle Position und Richtung, in Abhängigkeit davon können alle Zeichenoperationen durchgeführt werden. Hierzu haben wir detaillierte Beispiele im Skript beziehungsweise auf den Manualseiten:
Manualseite zu Plot5Streams
Manualseite zu Plotters
Manualseite zu TurtleGraphics
Außerdem die Beispiele und den bereitgestellten Programmrumpf beachten!

Was nun noch fehlt, ist der eigentlich Algorithmus zum Berechnen der Pflanze, die dargestellt werden soll.
Die Pflanze hat folgende Gestalt (der Algorithmus in Worten beschrieben):

Zunächst ein grader Strich nach oben.
Dann ein Strich nach links, einer nach rechts
Schliesslich zwei Striche senkrecht nach oben
Anschliessend wird jeder Ast nach obigem Muster ersetzt, und zwar bis zur angegebenen Tiefe (siehe Postscript-Dateien zu dieser Aufgabe bzw. die folgenden Abbildungen)
Update: leider war die Aufgabe nicht ganz eindeutig gestellt: wie aus den Bildern hervorgeht, sollten die Stengel (dazu gehoert alles, was nicht am Ende eines Astes ist) in jedem Schritt ihre Groesse verdoppeln.

Das ganze in Bildern (die einzelnen Bilder sind entsprechend verkleinert, das heißt, der Maßstab ist nicht ganz korrekt!) mit den Tiefen 1, 2, 3, 4 und 5:



Lange Rede ... Programmiert also ein kleines Oberon-Programm, dass folgende Punkte berücksichtigt:

- Es sollte eine Argumentverarbeitung vorliegen, so dass der Programmaufruf etwa folgendermaßen aussieht:
  programmname -X 5: Das Ergebnis sollte in einem Fenster mit einer Tiefe von 5 ausgegeben werden.
  programmname -o outputfile 5: Das Ergebnis wird in eine Datei namens *outputfile* geschrieben, wiederum mit einer Tiefe von 5.
  Werden keine Flags angegeben, sondern nur die Tiefe, soll das Ergebnis auf stdout ausgegeben werden. In allen anderen Fällen soll eine Usage-Meldung ausgegeben werden.
- Die eigentliche Berechnung der Pflanze gemäß des Algorithmus sollte rekursiv von statten gehen; das bedeutet, die berechnende Funktion ruft sich wiederholt selber auf. *Achtung:* Bei Rekursionen ist unbedingt auf eine geeignete Abbruchbedingung zu achten, so dass sich die Funktion nicht unendlich oft aufruft.
- Bitte die Beispiele, die in der Vorlesung vorgestellt wurden und die zu diesem Übungsblatt zur Verfügung gestellt werden, beachten; dort findet Ihr sicherlich einige wertvolle Tipps.

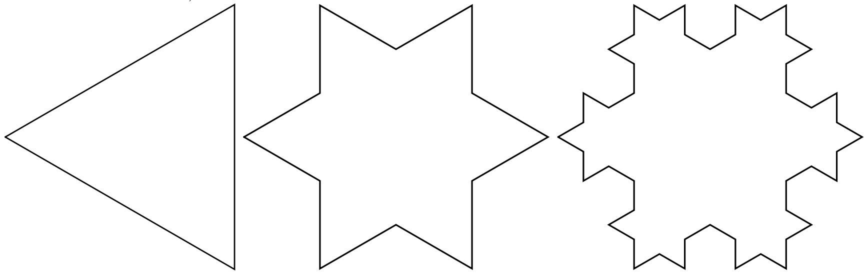
Allgemeine Tipps:
Um ein mit den obigen Modulen versehenes Programm zu übersetzen, benötigen wir das entsprechende Makefile. Dazu verwendet bitte den folgenden Befehl:
mmo -c makefile +l plot

1 Zusatzaufgabe: Leise rieselt der Schnee

Als kleine Zusatzaufgabe versuchen wir, eine Schneeflocke nachzubauen (allerdings eine sehr einfache) und grafisch wie in der ersten Aufgabe darzustellen.

Eine kleine Anleitung hierzu:
Ihr beginnt zunächst mit einem gleichseitigen Dreieck. Danach werden sukzessive alle Geraden ersetzt durch die Vorschrift, die sich aus den angefügten Postscript-Dateien bzw. Bildern ergibt, und zwar wiederum solange, bis die gewünschte Tiefe erreicht ist.

Bilder der Tiefen 1, 2 und 3:



Viel Spaß beim Knobeln!

Viel Erfolg!