Uebungsstunde vom 14.12.01, Info I

13. Dezember 2001

1 Abgabe/Korrekturen

- Per mail und auf Papier bis spätestens Freitag Abend (gleiche Version)
- Prozeduraufrufe und Output mit Copy-Paste unten hinkopieren
- Name des Autors im Programmcode
- Falls noch Compilierfehler: Fehlermeldungen dazukopieren
- ev. Traps auch ausdrucken

2 Nachbesprechung: Quicksort für Median/5% Berechnung

Finde die 15%-Grenze mittels Quicksort in diesem Array:

- Anfangszustand

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
7	29	2	70	14	18	68	32	50	74	4	19	81	27	46	72	11	23	52	75

- nach einem Schritt:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

⇒zwar nicht sortiert, aber sehr wohl getrennt!

3 Vorbesprechung: Rekursion

3.1 Was macht ein rekursives Programm aus?

- Eine Prozedur, die sich selber aufruft
- kompakter Programmcode
- klare Abbruchbedingungen
- vorsichtiger Umgang bei der Variablenübergabe (VAR oder nicht)
- mindestens eine Prozedur, die nicht rekursiv ist und die Rekursion startet

3.2 Beispiel Fakultätsberechnung

```
MODULE Fakultaet;      (* tb, 13.12.01, Demo für Rekursion     *)

        IMPORT In, Out;

        PROCEDURE calcRek(value: INTEGER):INTEGER;
                BEGIN
                        IF (value<=1) THEN
                                RETURN 1;
                        ELSE
                                RETURN (value *calcRek(value-1));
                        END;
        END calcRek;

        PROCEDURE start*();
                VAR result, value:INTEGER;
                BEGIN
                        result:=0;
                        In.Open; In.Int(value);
                        IF (In.Done) THEN
                                result:= calcRek(value);
                                Out.String("Die Fakultät von ");
                                Out.Int(value,4); Out.String(" ist ");
                                Out.Int(result,8); Out.Ln;
                        ELSE
                                Out.String("ProcedureCall:
                                        Fakultaet.start <integer> ");
                        END;
        END start;

END Fakultaet.


Fakultaet.start 7~
System.Free Fakultaet


Output:

Die Fakultät von   10 ist    24320
Die Fakultät von    7 ist     5040
```

3.3 Beispiel Pathfinder

Die Pathfindersonde beginnt in der linken oberen Ecke und legt den Weg in die rechte untere Ecke zurück. Sie kann sich nur nach unten und nach rechts bewegen. Finde den Weg, auf dem sie am meisten Punkte einsammeln kann.

4	7	1	4	10	7
8	8	7	3	5	2
1	7	6	7	8	5
4	6	5	9	2	8
4	2	7	6	1	10
7	6	6	6	2	4

- Wieviele mögliche Wege gibt es? (Grössenordnung) $\Rightarrow$ *Baum*

Rekursion probiert einfach alle möglichen Wege aus und wird deshalb auch oft brute force Methode genannt.

```
MODULE PathfinderRek;   (* tb, 13.12.01, Demo für Rekursion *)

        IMPORT Out, RN:=RandomNumbers;
        CONST size=6; skalierung=10;
        VAR feld: ARRAY size,size OF INTEGER; maxValue:INTEGER;

        PROCEDURE initFeld();
                VAR i,j:INTEGER;
                BEGIN
                        FOR i:=0 TO size-1 DO
                                FOR j:=0 TO size-1 DO
                                        feld[i,j]:=SHORT(ENTIER(RN.Uniform()
                                        *skalierung)+1);
                                END;
                        END;
        END initFeld;


        PROCEDURE show();
                VAR i,j:INTEGER;
                BEGIN
                        FOR i:=0 TO size-1 DO
                                FOR j:=0 TO size-1 DO
                                        Out.Int(feld[i,j],4);
                                END;
                                Out.Ln;
                        END;
                END show;
```

```
        PROCEDURE searchRek(
```

```
        PROCEDURE start*();
                BEGIN;
                     initFeld();
                     maxValue:=0;
                     searchRek(               );
                     show;
                     Out.String("Das Maximum ist ");
                     Out.Int(maxValue,8); Out.Ln;
        END start;

END PathfinderRek.


PathfinderRek.start~
System.Free PathfinderRek~

Output:

   4   7   1   4  10   7
   8   8   7   3   5   2
   1   7   6   7   8   5
   4   6   5   9   2   8
   4   2   7   6   1  10
   7   6   6   6   2   4
Das Maximum ist       75
```

4 Serie 6

4.1 Permutationen

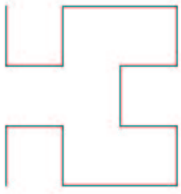
Möglichkeiten:

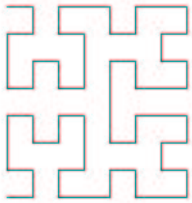
- Datentyp SET, INCL und EXCL zum Elemente hinzufügen/entfernen
- ARRAY OF BOOLEAN zum "Abhaken" der schon gebrauchten Elemente

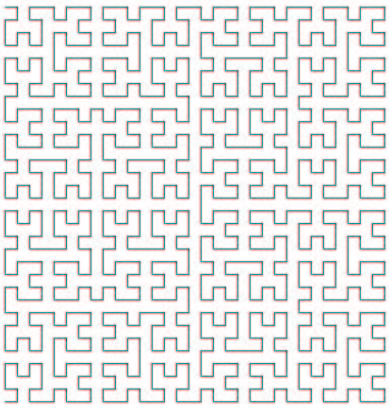
4.2 Hilbertkurve

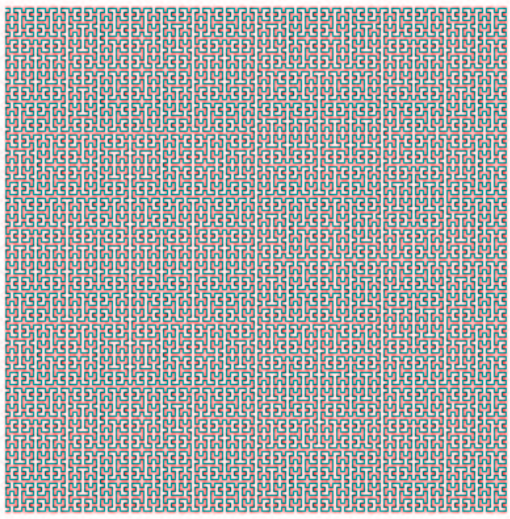
Turtle-Modul auf Website benutzen

1. Iteration: 

2.Iteration: 

3.Iteration: 

5. It: 

7. It: 

4.3 Gefangenenspaziergang

1.Tag:

1	2	3
4	5	6
7	8	9

2.Tag:

5	9	2
1	3	7
8	4	6

3.Tag: illegale Anordnung

2	4	7
3	6	2
8	5	9

4.4 Springerkreis

Erlaubte Züge: