

## Abitur 2002

### Aufgabe 1

(Aufgabentext / Lösungsvorschlag)

1.1

Ein Spiel wird von zwei Personen gespielt. Der eine Teilnehmer hat die Spielsteine X, der andere die Spielsteine O.

Die beiden Spieler setzen abwechselnd einen Spielstein auf ein Spielbrett, das aus drei mal drei Quadraten aufgebaut ist.

Das Spielbrett ist zu Beginn des Spieles leer.

Auf ein Quadrat darf nur ein Spielstein gesetzt werden.

Der Spieler, der als erster eine Zeile oder Spalte oder Diagonale mit seinen Spielsteinen besetzt, gewinnt das Spiel.

Das Spiel ist dann beendet.

In der folgenden Abbildung hat der Spieler mit den Spielsteinen X gewonnen:

X	O	O
O	X	
		X

Ein Computer soll das Setzen der Spielsteine durch die beiden Spieler während des gesamten Spieles verwalten. Der Computer soll absichern, dass keine zwei Spielsteine auf ein Quadrat gesetzt werden. Der Computer soll auch den Gewinner ermitteln.

Entwerfen Sie einen Algorithmus, der diese Aufgaben realisiert! Geben Sie eine Datenstruktur an, in der das Spielbrett gespeichert werden kann!

*VAR Spielfeld: ARRAY 4,4 OF CHAR;*

- 1. sinnvolle Anfangswerte:  
alle Felder erhalten als Startinhalt ein 'A'*
- 2. Ein Spieler darf auf ein Feld setzen, wenn er dran ist und der Inhalt des Feldes noch 'A' ist.*
- 3. Nach jedem Zug wird getestet, ob die drei Felder einer Zeile oder die drei Felder einer Spalte oder die drei Felder einer Diagonale einen einheitlichen Inhalt ungleich 'A' haben.*
- 4. In Abhängigkeit des Ergebnisses von 3. wird der Gewinner (X' oder 'O') ausgegeben.*

## 1.2

Gegeben ist eine Prozedur  $\text{Kreis}(x, y, r)$ , die einen Kreis mit dem Mittelpunkt  $M(x, y)$  und dem Radius  $r$  zeichnet. Die Parameter  $x, y$  und  $r$  sind ganze Zahlen.

Gesucht sind zwei Programme, die jeweils die gleiche Grafik auf den Bildschirm zeichnen. Die Grafik besteht aus konzentrischen Kreisen mit den Radien 1, 3, 5, ..., 99. Dem ersten Programm soll ein iterativer Algorithmus, dem zweiten ein rekursiver Algorithmus zu Grunde liegen.

Geben Sie die beiden Programme in Oberon oder Turbo Pascal an!

```
MODULE eins;  
  
PROCEDURE iterativ*;  
  
VAR x,y,r:INTEGER;  
  
BEGIN  
  FOR r :=1 TO 99 BY 2 DO  
    Kreis(100,100,r)  
  END;  
END iterativ;  
  
END eins.
```

```
MODULE zwei;  
  
PROCEDURE rekursiv(x,y,r:INTEGER);  
  
BEGIN  
  IF r<100 THEN  
    Kreis(x,y,r)  
    rekursiv(x,y,r+2);  
  END  
END rekursiv;  
  
PROCEDURE start*;  
  
BEGIN  
  rekursiv(100,100,1);  
END start,  
  
END zwei.
```

Vergleichen Sie Iteration und Rekursion! Beziehen Sie sich bei dem Vergleich auf die beiden Programme!

*Iteration ist Wiederholung durch Aneinanderreihung. Als Kontrollstrukturen werden Schleifen eingesetzt. In der Prozedur „iterativ“ des Modules „eins“ ist dies eine For-Schleife, die von 1 bis 99 läuft und bei der die Schleifenvariable nach jedem Durchlauf um zwei erhöht wird. Damit werden die Kreise mit den Radien 1 bis 99 nacheinander gezeichnet. Als einheitlicher Mittelpunkt wurde  $x=100$  und  $y=100$  gewählt.*

*Rekursion ist Wiederholung durch Ineinanderschachtelung. Als Kontrollstruktur wird hier meist die IF-THEN-ELSE-Verzweigung verwendet. In der Prozedur „rekursiv“ des Modules „zwei“ erfolgt zunächst der Aufruf der Zeichenprozedur Kreis( $x,y,r$ ) danach ruft sich die Prozedur „iterativ“ mit einem um zwei vergrößerten Radius selbst auf. Als einheitlicher Mittelpunkt wurde wiederum  $x=100$  und  $y=100$  gewählt.*

*Bemerkung:*

*Die noch nicht vollständig abgearbeiteten Unterprogramme werden auf einem Stapel inklusive aktueller Werte von Parametern und lokalen Variablen abgelegt. Das geschieht solange wie der Radius kleiner als 100 bleibt. Nach dem Erreichen der Austrittsbedingung – der Wert des Radius ist in diesem Fall auf 101 gewachsen – werden die Unterprogrammaufrufe in umgekehrter Reihenfolge beendet und vom Stapel entfernt.*