**C1 Vierertester**

**Ein Vierertester ist ein erkennender Endlicher Automat, der Dualzahlen darauf hin prüft, ob sie durch vier teilbar sind.**

1. **Gib eine Möglichkeit an, wie Dezimalzahlen in Dualzahlen umgewandelt werden können?**
2. **Wandle folgende Dezimalzahlen in Dualzahlen um!  
   a) 118  
   b) 56  
   c) 71**
3. **Was kennzeichnet eine durch vier teilbare Dualzahl? Wandle dazu mehrere durch vier teilbare Dezimalzahlen in Dualzahlen um und prüfe sie auf gemeinsame Eigenschaften!**
4. **Ergänze folgenden Zustandsgraphen für einen Automaten, der ein Vierertester ist!**
5. **Gib die Zustandstabelle an!**
6. **Prüfe mit Hilfe von Atocc folgende Wörter auf ihre Akzeptanz durch den Automaten!  
    a) 10010  
    b) 110100**

**Zusatz:   
Folgendes Strukturgramm zeigt eine Möglichkeit, Dezimalzahlen in Dualzahlen umzuwandeln. Setze es in ein Javaprogramm um!**

|  |  |
| --- | --- |
| **ganze Zahl: dezimal, rest** | |
| **Text: dual=“ “** | |
| **mache**  **solange (dezimal>0)** | **rest=dezimal%2** |
| **dual=rest+dual** |
| **dezimal=dezimal/2** |
|  |

**Lösung**

Zu 1.)

Divisionsmethode

Zerlegungsmethode

Zu 2.)

Bsp für 118

56

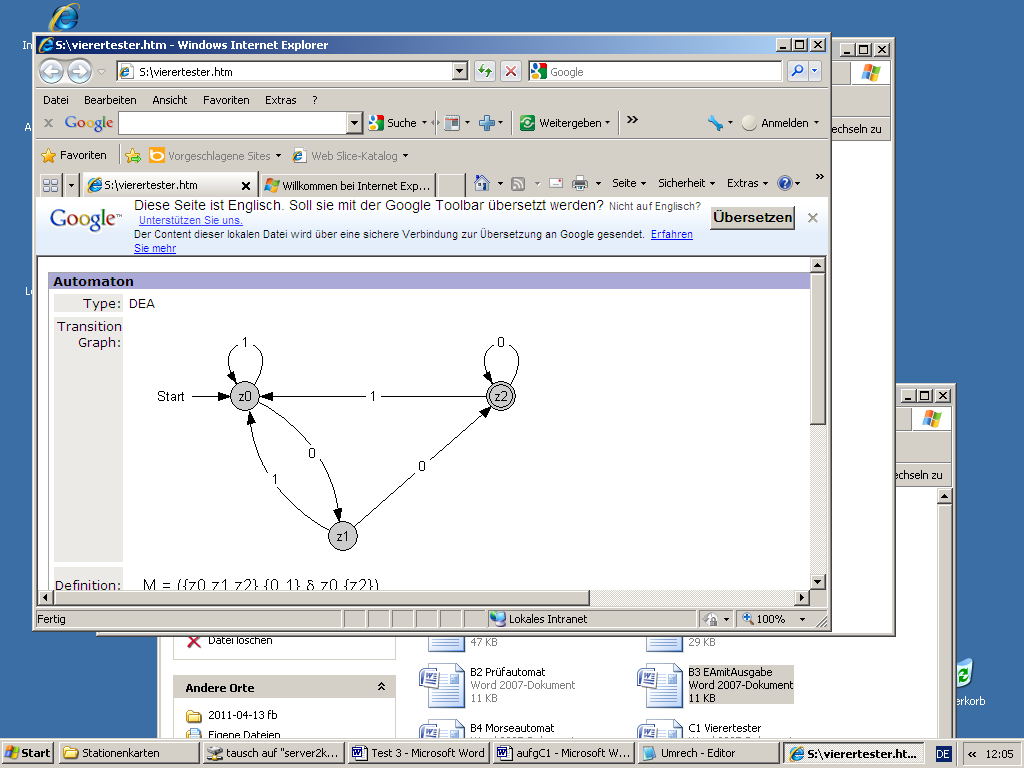
71

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 26 | 25 | 24 | 23 | 22 | 21 | 20 |
| 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 |
|  | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 |

Zu 3.)

Die letzten beiden Bit sind 0.

Zu 4.)



Zu 5.)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Z0 | Z1 | Z2 |
| 0 | Z1 | Z2 | Z2 |
| 1 | Z0 | Z0 | Z0 |

Zu 6.)

10010 wird nicht akzeptiert

110100 wird akzeptiert

Beweis mit Atocc

Zusatz:

String dual="";

do {

int rest=dezimal%2;

dual=String.valueOf(rest)+dual;

dezimal=dezimal/2;

}

while (dezimal>0);