

Einführung

Im Land der Matsedanier herrscht großer Frust. Seit Jahren haben sie Probleme mit der Anbindung an das Internet, da es in ihrem Land bisher leider keine flächendeckende Ausstrahlung mit Funk gab. Der Grund dafür ist, dass das Land der Matsedanier leider auch nicht ebenerdig sondern sehr hügelig ist, mit Höhenunterschieden von bis zu mehreren hundert Metern.

In der Abbildung 1 sieht man einen Ausschnitt des Landes von Oben. Jedes Quadrat ist 100 x 100 Meter groß und hat an den Eckpunkten die jeweilige Höhenangabe in 100 Meter.

In dem Beispiel des Quadrates Q_3 haben die Eckpunkte P_1 , P_2 , P_3 und P_4 des Quadrates die Höhe 100, 200, 300 und 400 Meter (siehe auch Abbildung 2 – Beispiel Quadrat Q_3).

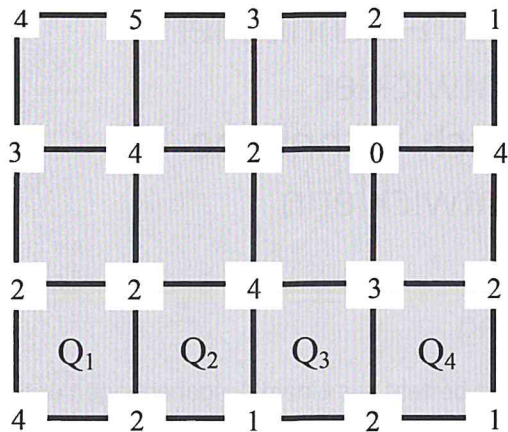


Abbildung 1 – Ansicht von Oben mit Höhenangaben der Ecken und Kreuzungen

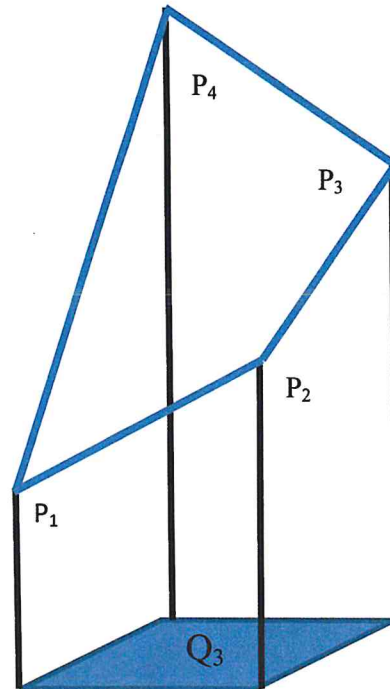


Abbildung 2 – Beispiel Quadrat Q_3

Um jetzt flächendeckendes Internet zur Verfügung zu stellen, muss das ganze Land möglichst gut ausgeleuchtet werden. Dazu sollte an den jeweiligen Eckpunkten der Quadrate jeweils eine Antenne aufgestellt werden. Leider haben die Matsedanier nicht mit den enormen Kosten gerechnet, wenn man diese erste Idee umsetzen würde. Auch hat man erkannt, dass man dann viel zu viele Antennen hätte, die man evtl. gar nicht bräuchte, da diese dann das gleiche Gebiet doppelt und dreifach abdecken. Leider haben die Matsedanier keine Idee, wo man sinnvoller Weise die Antennen platziert, um mit möglichst wenig Antennen das komplette Land auszuleuchten.

Daher haben die Matsedanier die Firma Matse AG mit einer Lösung des Problems beauftragt, d. h. Sie sollen ein Programm schreiben, mit dem für ein vorgegebenes Gelände die **minimale** Anzahl von Antennen und deren Position bestimmt werden kann.

Jede Antenne hat eine Höhe von zehn Metern und kann an jedem Eckpunkt eines Quadrates stehen. D. h. bei einem Feld wie in Abbildung 1 gibt es maximal 20 mögliche Positionen und maximal 20 Antennen, die man platzieren könnte.

Die Höhe des Geländes ist zwischen 0 Metern und maximal 600 Meter. Die Angaben der Höhen erfolgt durch ganze Zahlen von 0 bis 6, wobei die 0 für null Meter und die 6 dann für 600 Meter steht.

Wann ist ein Quadrat durch eine Antenne erreichbar?

Um zu überprüfen, ob ein Feld abgedeckt wird, ist es ausreichend zu testen, ob die Eckpunkte des Quadrates von der Antenne durch eine direkte Verbindung zu erreichen sind. Es ist nicht gefordert, die durch die Punkte aufgespannte Fläche zwischen den Eckpunkten zu betrachten, da die vier Eckpunkte in der Regel nicht in einer Ebene liegen müssen und die möglichen Teilebenen durch je drei Eckpunkte nicht eindeutig sind (siehe auch Abbildung 3 – Nicht eindeutige Bildung von Teilflächen durch je drei Punkte)

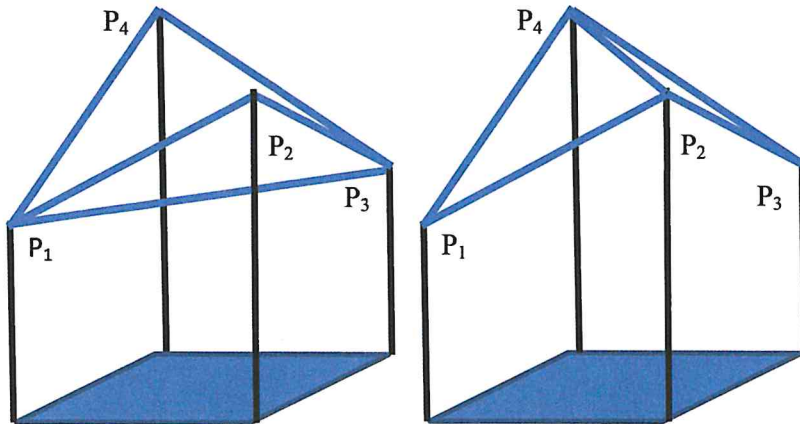


Abbildung 3 – Nicht eindeutige Bildung von Teilflächen durch je drei Punkte

Ein Quadrat Q_i ist abgedeckt bzw. durch eine oder mehrere beliebige Antennen erreichbar, wenn alle Eckpunkte des Quadrates Q_i durch einen Strahl von diesen Antennen zu den jeweiligen Eckpunkten erreichbar sind. In Abbildung 4 ist der Eckpunkt P_1 nicht durch die Antenne bei P_6 erreichbar, da die Strecke von der Antennenspitze zu P_1 die Fläche des Trapezes P_2, P^*_2, P^*_5 und P_5 bei S durchstößt.

Das bedeutet, dass es mindestens eine Verbindung von einer beliebigen Antennenspitze zu jedem der Eckpunkte geben muss.

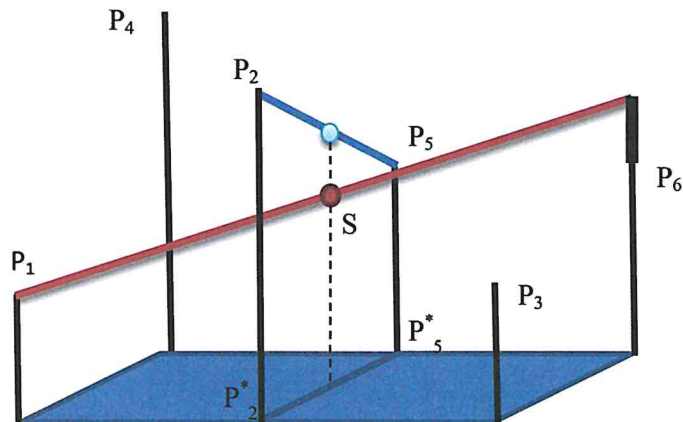


Abbildung 4 – P_1 ist von der Antenne bei P_6 nicht mehr erreichbar

In einem weiteren Beispiel mit der Abbildung 5 steht die Antenne bei P_8 . Damit sind die Punkte P_1 und P_5 nicht zu erreichen, da das Trapez der Punkte P_2 und P_6 mit ihren Bodenpunkten P^*_2 und P^*_6 den Strahl nicht durchlässt. Das bedeutet, dass die Antenne bei P_8 unglücklich gestellt ist oder noch eine weitere Antenne benötigt wird.

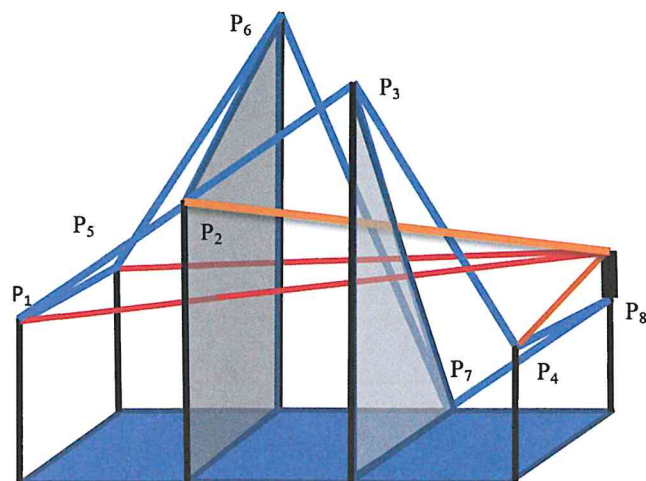
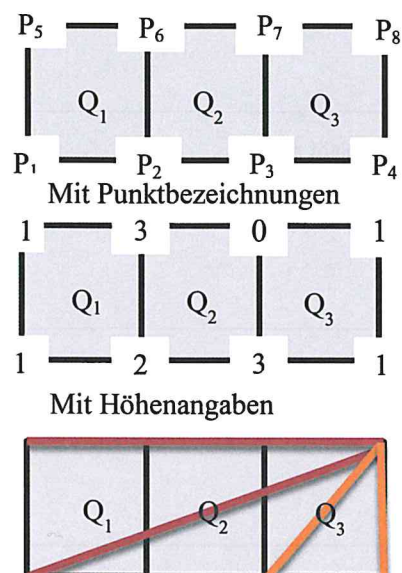


Abbildung 5 – Beispiel mit einer Antenne bei P_8



Eingabeformat

Die Eingabe der Testfälle erfolgt durch eine Datei, die zu Beginn des Programms eingelesen wird. Dazu wird der Name der Datei als Parameter beim Programmaufruf mit übergeben.

Beschreibung der Testfälle

Die Eingabe der Höhen erfolgt durch die übergebene Datei. Die Beschreibung einer Eingabedatei besteht aus einer Zeile. Sie steht immer am Anfang dieser Datei und wird in Kommentarzeilen, wie unten gezeigt, angegeben. Jede Kommentarzeile beginnt mit einem Semikolon. Die Beschreibung ist bei der Ausgabe des Ergebnisses mit auszugegeben.

```
;*****  
;  Testbeispiel von Abbildung 1  
;*****
```

Definition der Größe eines Gebietes

Die Fläche des Gebietes ist immer rechteckig und wird durch die Kantenlängen des Rechtecks definiert. Die Eingabe erfolgt durch ganzzahlige Koordinaten. Eine Einheit entspricht dabei einhundert Metern.

```
;  Groesse des Gebietes  
5 4
```

Definition der Höhen

Die Höhen einer jeweiligen Zeile werden hintereinander geschrieben. Das nachfolgende Beispiel entspricht damit dem Feld von Abbildung 1.

```
;  Hoehen  
4 5 3 2 1  
3 4 2 0 4  
2 2 4 3 2  
4 2 1 2 1
```

Sollten weniger Werte in einer Zeile stehen als die Dimension des Feldes, so sind die restlichen Höhen mit dem letzten gelesenen Wert der jeweiligen Zeile zu versehen.

In jeder Zeile ist jedoch mindestens ein Wert anzugeben.

Es gibt keine negativen Höhenangaben.

Beispiel 1 komplett:

```
;*****  
;  Testbeispiel von Abbildung 1  
;*****  
;  Groesse des Gebietes  
5 4  
;  Hoehen  
4 5 3 2 1  
3 4 2 0 4  
2 2 4 3 2  
4 2 1 2 1
```

Weitere zulässige Eingaben

Weitere zulässige Angaben zu der Größe 5 4 wären unter Berücksichtigung der Dimension

```
;  Hoehen  
4  
3  
2  
4
```

Diese Eingabe ist äquivalent zu

```
;  Hoehen  
4 4 4 4 4  
3 3 3 3 3  
2 2 2 2 2  
4 4 4 4 4
```

Das nachfolgende Beispiel

```
;  Hoehen  
4 2 1 0  
3 1 5  
2  
4 1 2
```

wird zu dem nachfolgenden Feld erweitert.

```
;    Hoehen
4 2 1 0 0
3 1 5 5 5
2 2 2 2 2
4 1 2 2 2
```

Ausgabeformat

Die Ausgabe erfolgt in zwei Teilen:

1. Ausgabe am Monitor und in Datei
2. Dateien für Gnuplot

Ausgabe am Monitor und in Datei

Am Monitor werden die folgenden Informationen ausgegeben:

- die eingelesene Beschreibung des Testfalles,
- die Ausdehnung in X- und Y-Richtung (Dimension des eingelesenen Feldes),
- die Höhenangaben des eingelesenen Feldes, die Anzahl der minimal benötigten Antennen und deren Positionen.

Zusätzlich werden diese Daten in eine Datei geschrieben.

Beispielausgabe zu dem Beispiel aus Abbildung 1

```
*****
Testbeispiel von Abbildung 1
*****
Ausdehnung in X: 5
Ausdehnung in Y: 4
Eingelesene Höhen
4 5 3 2 1
3 4 2 0 4
2 2 4 3 2
4 2 1 2 1
Benötigte Antennen: 1
Antenne 1: 2 2
```

Bei der Angabe der Antennenposition ist lediglich die Koordinate in X Y anzugeben. Die Höhe der Antenne ist hier nicht notwendig. Sollten mehrere Antennen benötigt werden, können diese in beliebiger Reihenfolge ausgegeben werden.

Ausgabe für Gnuplot

Für die bessere Überprüfung des Ergebnisses soll dieses unter zur Hilfenahme von Gnuplot dargestellt werden. Dazu sind für jeden Test die folgenden Dateien zu bilden:

- *TestX.dem*
Enthält die eigentliche Definition, was zu zeichnen ist. Das X ist durch die entsprechende Nummer zu ersetzen.
- *TestX.dat*
Die Dateien enthalten die Koordinaten des eingelesenen Feldes.
Dabei werden die jeweiligen Koordinaten einer Reihe untereinander geschrieben
Zwischen den Datenreihen wird eine Leerzeile eingefügt (siehe auch das Beispiel weiter unten).
- *TestXAntenneY.dat*
Die Dateien enthalten die Koordinaten einer Antenne. Dabei sind zwei Zeilen notwendig: eine für den Eckpunkt und eine für den um 0.1 (Höhe der Antenne) erhöhten Eckpunkt.

Testbeispiel von Abbildung 1

Test1.dem

```
# Aufgabe 1 vom Aufgabenblatt

set title "Aufgabe 1"
set hidden3d
set key outside top
splot 'Test1.dat' w line , \
      'Test1Antenne1.dat' w linesp
pause -1 "Hit return to continue"
```

Test1.dat

```
0 0 4
0 1 5
0 2 3
0 3 2
0 4 1
1 0 3
1 1 4
1 2 2
1 3 0
1 4 4
2 0 2
2 1 2
2 2 4
2 3 3
2 4 2
3 0 4
3 1 2
3 2 1
3 3 2
3 4 1
```

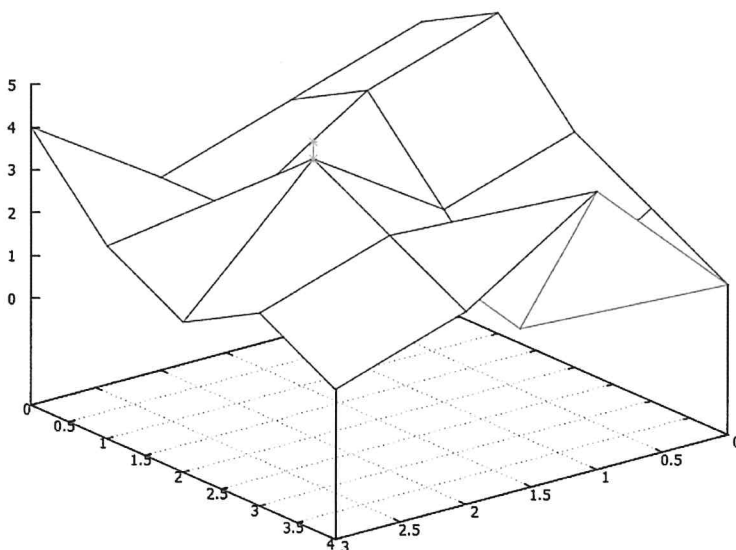
Test1Antenne1.dat

```
2 2 4
2 2 4.1
```

Ergebnis mit Gnuplot

Aufgabe 1

'Aufgabe1.dat' —
'Aufgabe1Antenne1.dat' —*



Weitere Erklärungen zur Datei Test1.dem.

Die einzelnen Zeilen der Datei haben die folgende Bedeutung:

`set title "Aufgabe 1"`

Setzt den Titel der Ausgabe

`set hidden3d`

Kanten, die durch andere Flächen verdeckt sind, werden nicht dargestellt.

`set key outside top`

Die Beschriftung erfolgt außerhalb der Zeichnung.

`splot 'Test1.dat' w line , \`
`'Test1Antenne1.dat' w linesp`

Der Befehl `splot` zeichnet das Gelände, dabei werden mehrere Dateien untereinander angegeben. Die Zeilen werden mit einem Komma und einem Backslash getrennt. Es können so beliebig viele Dateien kombiniert ausgegeben werden.

Die Parameter `line` bzw. `linesp` geben an, dass die eingelesenen Punkte mit Linien verbunden werden. Zusätzlich wird bei `linesp` noch ein Punkt an die eingelesenen Koordinaten gesetzt.

`pause -1 "Hit return to continue"`

Das Programm wartet auf eine Eingabe, bevor das Fenster geschlossen wird.

Weitere Beispiele

Beispiel 2 – Eingabe

```
;*****  
; Beispiel 2  
;*****  
; Groesse des Gebietes  
5 4  
; Hoehen  
1 3 2 4 3  
5 4 2 2 0  
3 2 4 3 1  
2 0 3 2 3  
1 4 2 1 2  
0 3 2 2 0
```

Beispiel 2 – Ausgabe

```
*****  
Beispiel 2  
*****  
Ausdehnung in X: 5  
Ausdehnung in Y: 6  
Eingelesene Höhen  
1 3 2 4 3  
5 4 2 2 0  
3 2 4 3 1  
2 0 3 2 3  
1 4 2 1 2  
0 3 2 2 0  
Benötigte Antennen: 3  
Antenne 1: 0 1  
Antenne 2: 2 2  
Antenne 3: 3 0
```

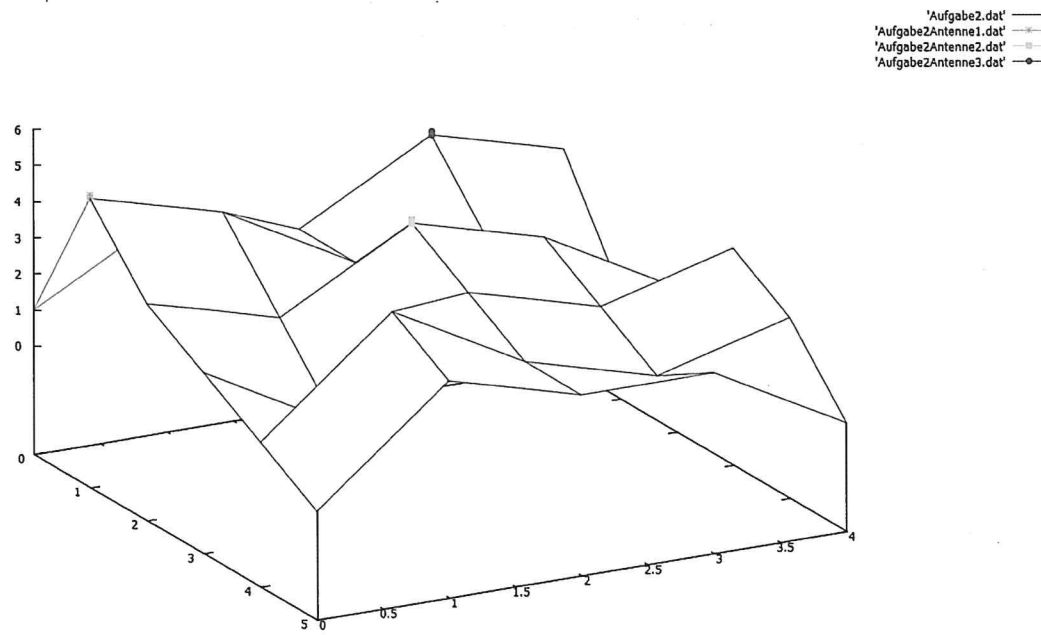
Beispiel2.dem

```
# Beispiel2  
set title "Beispiel 2"  
set hidden3d  
set key outside top  
splot 'Test2.dat' w line , \  
      'Test2Antenne1.dat' w linesp , \  
      'Test2Antenne2.dat' w linesp , \  
      'Test2Antenne3.dat' w linesp  
pause -1 "Hit return to continue"
```

Auf die Angabe der Dateien Beispiel2.dat, Beispiel2Antenne1.dat, ... wird hier aus Platzgründen verzichtet.

Ausgabe mit Gnuplot

Beispiel 2



Beispiel 3

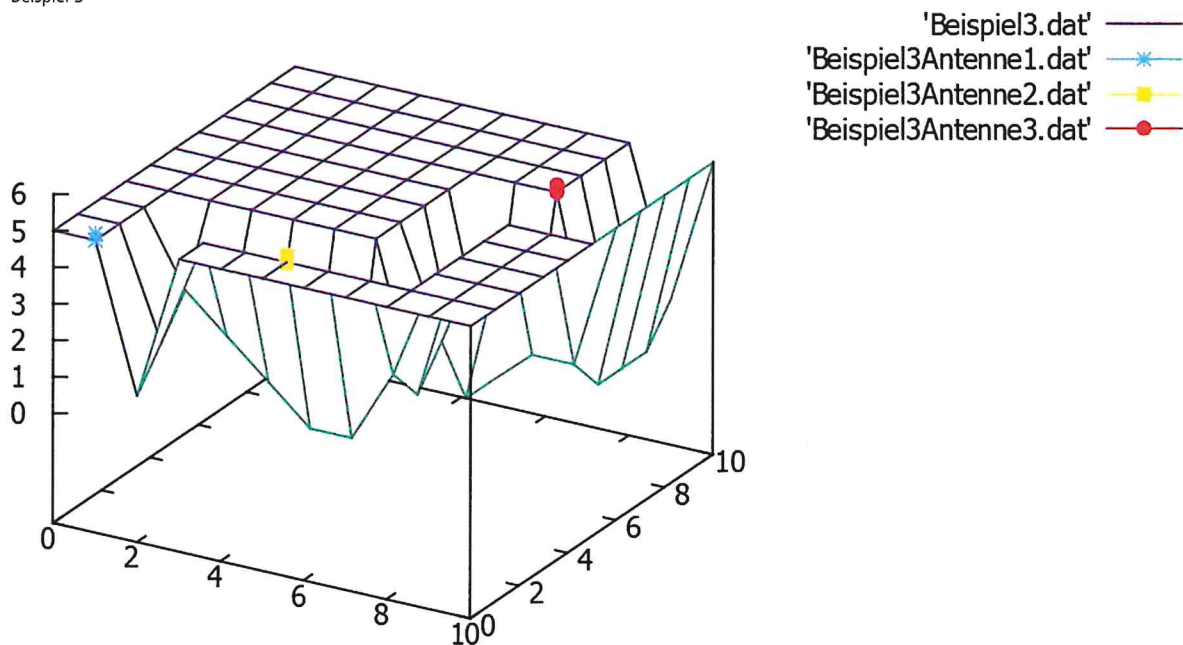
Beispiel 3 – Eingabe

```
;*****  
; Beispiel 3  
;*****  
; Groesse des Gebietes  
11 11  
; Hoehen  
5  
5  
1 2 3 5  
5 5 2 5  
5 5 1 5  
5 5 0 5  
5 5 0 5  
5 5 2 1 2 0 0 5  
5 5 5 5 5 5 1 5  
5 5 5 5 5 5 1 0 0 0 1  
5  
5
```

Beispiel 3 – Ausgabe

```
*****  
Beispiel 3  
*****  
Ausdehnung in X: 11  
Ausdehnung in Y: 11  
Eingelesene Höhen  
5 5 5 5 5 5 5 5 5 5 5  
5 5 5 5 5 5 5 5 5 5 5  
1 2 3 5 5 5 5 5 5 5 5  
5 5 2 5 5 5 5 5 5 5 5  
5 5 1 5 5 5 5 5 5 5 5  
5 5 0 5 5 5 5 5 5 5 5  
5 5 0 5 5 5 5 5 5 5 5  
5 5 2 1 2 0 0 5 5 5 5  
5 5 5 5 5 5 1 5 5 5 5  
5 5 5 5 5 5 1 0 0 0 1  
5 5 5 5 5 5 5 5 5 5 5  
5 5 5 5 5 5 5 5 5 5 5  
Benötigte Antennen: 3  
Antenne 1: 0 1  
Antenne 2: 1 5  
Antenne 3: 7 8
```

Beispiel 3



Im Rahmen der schriftlichen Aufgabe sind am ersten Tag abzugeben:

- Aufgabenanalyse und Verfahrensbeschreibung zur Ermittlung der minimalen Anzahl und der Position der Antennen
 - Beschreibung der Aufgabe, d. h. Analyse der Problemstellung sowie Diskussion einer geeigneten Programm- und Datenstruktur für das angegebene Problem
 - Einlesen und Initialisieren der Daten
 - Entwurf des Algorithmus inklusive der Beschreibung der Verfahren zur Ermittlung der minimalen Anzahl und der Position der Antennen
 - Ausgabe gemäß Aufgabenstellung
- Programmkonzeption unter Berücksichtigung der funktionalen Trennung
 - Klassen, Methoden und Datenstrukturen in Form von UML-Diagrammen
 - UML-Sequenzdiagramme für die wesentlichen Abläufe
 - Detaillierte Beschreibung der wesentlichen Methoden in Form von Nassi-Shneiderman-Diagrammen oder Programmablaufplänen
- Kommentierte Simulation eines Beispiels zum Finden der minimalen Anzahl von Antennen und der Bestimmung der Position jeder Antenne (Schreibtischtest)

Im Rahmen des Prüfprodukts sind in elektronischer Form (PDF-Dokument) abzugeben:

- Verbale Beschreibung des realisierten Verfahrens mit Beispiel (Schreibtischtest)
- Programmsystem (bestehend aus Klassen, Methoden) als Quellcode
- Entwicklerdokumentation
- Benutzeranleitung
- Ausführliche Beschreibung, Begründung und Diskussion
 - der angegebenen Beispiele und
 - einer ausreichenden Zahl von Beispielen mit Ein- und Ausgabe, die sowohl die Normalfälle als auch auftretende Spezial- und Fehlerfälle abdecken
- Zusammenfassung und Ausblick (z. B. Erweiterungsmöglichkeiten)

Zusätzlich sind im Rahmen des Prüfprodukts in elektronischer Form abzugeben:

- Programmsystem in ausführbarer Form
- Ein- und Ausgabedateien Ihrer Beispiele
- Ein Skript zur automatischen Ausführung aller von Ihnen angegebenen Beispiele

Dem Prüfprodukt ist eine eigenhändig unterschriebene Eigenständigkeitserklärung (Juristische Erklärung) folgenden Inhalts beizufügen:

„Ich erkläre verbindlich, dass das vorliegende Prüfprodukt von mir selbstständig erstellt wurde. Die als Arbeitshilfe genutzten Unterlagen sind in der Arbeit vollständig aufgeführt. Ich versichere, dass der vorgelegte Ausdruck mit dem Inhalt des von mir erstellten Datenträgers identisch ist. Weder ganz noch in Teilen wurde die Arbeit bereits als Prüfungsleistung vorgelegt. Mir ist bewusst, dass jedes Zuwiderhandeln als Täuschungsversuch zu gelten hat, der die Anerkennung des Prüfprodukts als Prüfungsleistung ausschließt.“

Im Rahmen des auftragsbezogenen Fachgesprächs sind die Aufgabenanalyse und der Lösungsentwurf zu begründen und das Prüfungsprodukt zu erläutern.