

## Übung 8

Arbeitsaufwand insgesamt: 10h

### Inhaltsverzeichnis

Übung 6 .....	1
Teil 1 – „Über die Ausbreitung von Gerüchten und die Fehler, die einem bei der Simulation dessen passieren können“ .....	2
Lösungsidee: .....	2
Lösung: .....	3
Testfälle: .....	4
Fehlerliste: .....	9

## Teil 1 – „Über die Ausbreitung von Gerüchten und die Fehler, die einem bei der Simulation dessen passieren können“

Ziel ist es mithilfe des bereitgestellten Dokuments „Gerüchte und Vehler“ einen Algorithmus zu entwerfen, der die Ausbreitung eines Gerüchtes für eine gegebene Bevölkerungsanzahl simuliert. Die Ergebnisse sollen zudem in ein File geschrieben werden, um diese in Excel visualisieren zu können. Außerdem soll der Zustand der Bevölkerung am Ende des Programms auf der Konsole ausgegeben werden.

### Lösungsidee:

Der User soll sich aussuchen können, wie groß die Population ist und welche Methode verwendet werden soll.

Für die Simulation müssen sich jeweils zwei zufällige Personen solange treffen, bis es keinen Verbreiter mehr gibt. Die gewählten Personen dürfen auch nicht ident sein (nicht die gleiche Person).

Wurden die beiden Personen gewählt, so muss deren Rolle bestimmt werden. Also, ob die Person ein Nichtwissender (Nitwit), Verbreiter (Propagator) oder Unterdrücker (Suppressor) ist. Dies kann entweder über einen Vektor mit Rollen geschehen, wobei der Wert an der Stelle der bestimmten Zufallszahl die Rolle ist.

Alternativ kann die Rolle auch bestimmt werden, indem überprüft wird ob die Zahl in einem gewissen Bereich ist. Ist die Zahl z.B. kleiner als die Anzahl der Nichtwissenden, so ist es ein Nichtwissender. Ist die Zahl größer als die Anzahl der Nichtwissenden, aber kleiner als die Anzahl der Nichtwissenden UND der Verbreiter, so ist es ein Verbreiter. Alles darüber, bis zum Maximum sind die Unterdrücker. Das reduziert auch die benötigten Ressourcen.

Interessant sind hier aber vor allem die Begegnungen, bei denen mindestens eine Person ein Verbreiter ist, weil hier entweder Nichtwissende zu Verbreitern oder Verbreiter zu Unterdrückern werden. Das sind 5 der 9 möglichen Fälle. Unabhängig davon muss aber jede Begegnung ins File geschrieben werden.

Zudem gibt es mehrere Methoden: Die von Daley-Kendall und jene von Maki-Thompson.

1. Bei der Methode von Daley-Kendall werden bei Verbreiter-Verbreiter treffen BEIDE Personen zu Unterdrückern.
2. Bei der Methode von Maki-Thompson wird bei Verbreiter-Verbreiter treffen NUR EINE Person zu einem Unterdrücker.

Durch das beigelegte PDF entstanden bei mir leider Verwirrungen bezüglich der genauen Implementation: Während die Grafiken alle darauf hinweisen, dass beide Personen komplett zufällig ausgewählt werden, schreibt der Autor mehrmals, dass eine Person aus dem Pool der Verbreiter entnommen wird bzw. immer mind. eine Person ein Verbreiter ist. Zudem schrieb der Autor ganz am Schluss, dass nun doch beide Methoden einen Theta-Wert von 0,203 erreichen sollten: Konträr zu den anfänglichen 0,283 für Daley-Kendal und den 0,203 für Maki. Weshalb ich diese Werte als mein Ziel setze.

Das Programm endet, sobald die Anzahl der Verbreiter oder Nichtwissenden auf 0 sinkt, wobei zweiteres zwar sehr, sehr unwahrscheinlich, aber dennoch möglich ist.

Danach kann die Datei, sofern diese weniger als 1.048.576 Zeilen hat, eingelesen werden und ein Liniendiagramm daraus erstellt werden.

Anmerkungen nach Entwicklung des Programms:

Der Wert, von dem der Autor spricht, nämlich ca. 0,203 für das Maki-Thompson Modell, wird nur erreicht, wenn immer eine zufällig ausgewählte Person ein Verbreiter ist. Dadurch ergibt sich aber ein anderes Diagramm, welches wesentlich flacher verläuft. Aus diesem Grund habe ich alle 4 Versionen implementiert: Jeweils 1x die Methoden mit komplett zufälliger Auswahl und jeweils 1x wobei immer eine Person ein Verbreiter ist.

Zudem habe ich sowohl die Version mit dem Vektor zur Bestimmung, sowie die Version mit dem Bereich implementiert, weil beim Testen die oben erwähnten Unstimmigkeiten aufgetreten sind. Schlussendlich habe ich mich entschieden die Version mit dem Bereich zu verwenden. Eine Auswahl bzw. Unterbringung in ein einziges Programm würde nur wenig Sinn machen, wenn diese nur die Performance beeinträchtigen würde. Diese Version findet sich aber immer noch auskommentiert in der Datei „Alternative.cpp“. Beide Versionen erwähnt der Autor auch in seinem Artikel.

Lösung:

Als C++ Projekt „Teil\_1“ im Archiv

## Testfälle:

- Falsche Eingaben bei Populationsabfrage
- Falsche Eingaben bei Methodenbestimmung
- Fehlerfreie Eingaben

*Falsche Eingaben bei Populationsabfrage:*

Nicht numerische Strings bei Populationsabfrage:

```
Please enter the population size: a10
The value must be numeric (MAX:2147483647): @10
The value must be numeric (MAX:2147483647): ,10
The value must be numeric (MAX:2147483647): 10000a1
Do you want to use the Daley or Maki method? (D/M): D
Do you want one person to ALWAYS propagate? (Y/N): Y
nitwits:          28.1%
propagators:      0.0%
suppressors:      71.9%
```

Das Problem mit den nachfolgenden Zeichen konnte ich nach wie vor nicht lösen, beeinflusst aber die Funktionalität nicht. Ich habe alternativ probiert, die Zahl als String einzulesen, über Regex Filter zu überprüfen und danach umzuwandeln. Hierbei stellt sich aber das Problem, dass Zahlen größer als das Integer-Limit bei Umwandlung einen Anwendungsabsturz verursachen.

Zu große Zahl (übersteigt unsigned Int):

```
Please enter the population size: 289239812393312
The value must be numeric (MAX:2147483647): 2147483648
The value must be numeric (MAX:2147483647): 10000
Do you want to use the Daley or Maki method? (D/M): D
Do you want one person to ALWAYS propagate? (Y/N): Y
nitwits:          28.1%
propagators:      0.0%
suppressors:      71.9%
```

Zu kleine Zahl (Kleiner gleich 0):

```
Please enter the population size: -1
Entered size was too low. Terminating.
```

*Falsche Eingaben bei Methodenbestimmung:*

Strings welche sich von den vorgegebenen Möglichkeiten unterscheiden:

```
Please enter the population size: 10000
Do you want to use the Daley or Maki method? (D/M): A
Do you want to use the Daley or Maki method? (D/M): 1
Do you want to use the Daley or Maki method? (D/M): ,
Do you want to use the Daley or Maki method? (D/M): M
Do you want one person to ALWAYS propagate? (Y/N): D
Do you want one person to ALWAYS propagate? (Y/N): K
Do you want one person to ALWAYS propagate? (Y/N): 1293
Do you want one person to ALWAYS propagate? (Y/N): @
Do you want one person to ALWAYS propagate? (Y/N): y
nitwits:          19.9%
propagators:      0.0%
suppressors:      80.1%
```

Groß- und Kleinschreibung ist irrelevant:

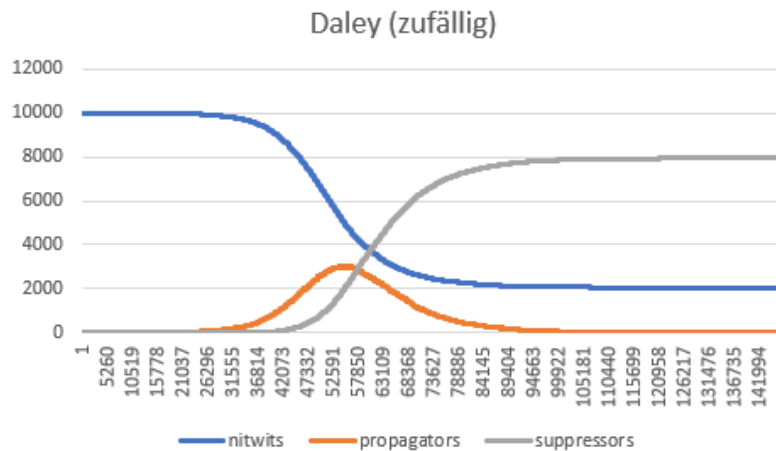
```
Please enter the population size: 10000
Do you want to use the Daley or Maki method? (D/M): m
Do you want one person to ALWAYS propagate? (Y/N): Y
nitwits:          20.9%
propagators:      0.0%
suppressors:      79.1%
```

*Fehlerfreie Eingaben:*

Populationsgröße 10.000 – Daley (beide zufällig):

```
Please enter the population size: 10000
Do you want to use the Daley or Maki method? (D/M): d
Do you want one person to ALWAYS propagate? (Y/N): n
nitwits:      20.5%
propagators:  0.0%
suppressors:  79.5%
```

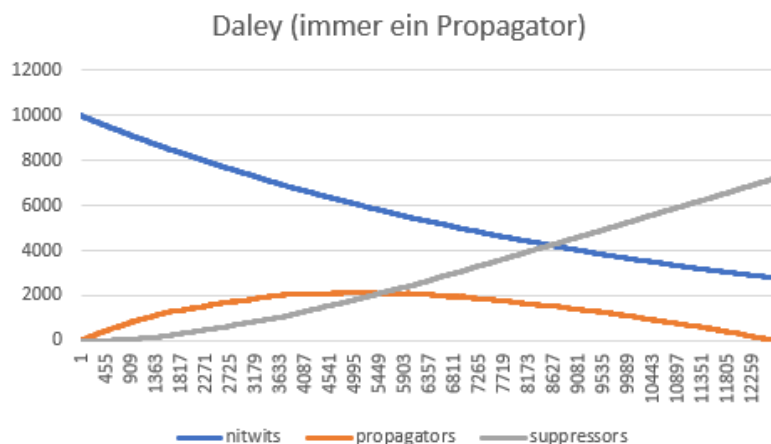
Das gewünschte Ergebnis wäre 20,3%, bei der geringen Population von 10000 kann es aber durchaus eine kleine Abweichung geben. 20,6% liegt dem Ziel aber sehr nahe.



Populationsgröße 10.000 – Daley (immer ein Propagator):

```
Please enter the population size: 10000
Do you want to use the Daley or Maki method? (D/M): Daley
Do you want one person to ALWAYS propagate? (Y/N): y
nitwits:      27.8%
propagators:  0.0%
suppressors:  72.2%
```

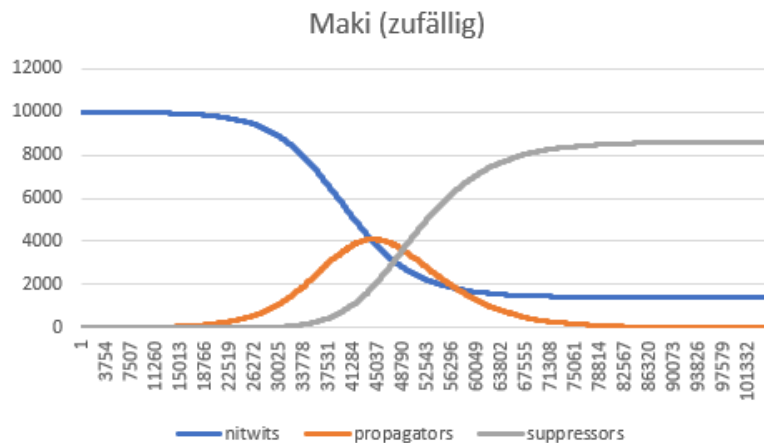
Hierfür wäre das Zielergebnis 28,3%. Wiederum ist eine kleine Abweichung der Fall.



Populationsgröße 10.000 – Maki (beide zufällig):

Please enter the population size: 10000  
 Do you want to use the Daley or Maki method? (D/M): M  
 Do you want one person to ALWAYS propagate? (Y/N): n  
 nitwits: 13.8%  
 propagators: 0.0%  
 suppressors: 86.2%

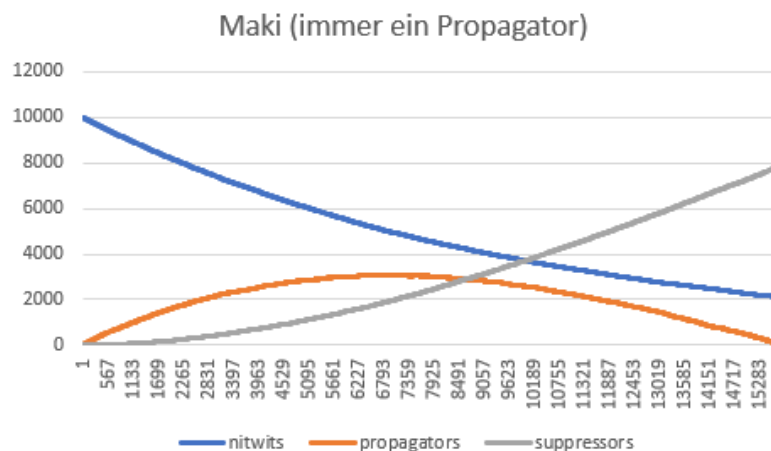
Diesen Wert hat der Autor nicht erwähnt, weil es anscheinend die „falsche“ Art ist die Maki Methode anzuwenden. Es würde aber den Grafiken entsprechen. Dieser Theta-Wert befindet sich ca. bei 0,130 also 13%.



Populationsgröße 10.000 – Maki (immer ein Propagator):

Please enter the population size: 10000  
 Do you want to use the Daley or Maki method? (D/M): M  
 Do you want one person to ALWAYS propagate? (Y/N): y  
 nitwits: 20.6%  
 propagators: 0.0%  
 suppressors: 79.4%

Anschließend ist hier das „richtige“ Maki Modell mit einem Nitwit Anteil von 20,6%, also leicht abweichend von den 20,3%.



Anschließend wollte ich noch sichergehen, dass sich bei hohen Populationswerten der Wert immer weiter den Theta-Werten annähert. Hierzu gibt es leider keine Excel-Diagramme, weil die Dateien zu groß wurden und dementsprechend auch zu viele Zeilen hatten.

output.csv 11.01.2020 19:53 Microsoft Excel-C... 302.845 KB

Populationsgröße 1.000.000 – Daley (beide zufällig):

```
Please enter the population size: 1000000
Do you want to use the Daley or Maki method? (D/M): D
Do you want one person to ALWAYS propagate? (Y/N): n
Please wait! Based on the entered number this can take a while...
nitwits:          20.4%
propagators:      0.0%
suppressors:      79.6%
```

Populationsgröße 1.000.000 – Daley (immer ein Propagator):

```
Please enter the population size: 1000000
Do you want to use the Daley or Maki method? (D/M): D
Do you want one person to ALWAYS propagate? (Y/N): Y
Please wait! Based on the entered number this can take a while...
nitwits:          28.5%
propagators:      0.0%
suppressors:      71.5%
```

Populationsgröße 1.000.000 – Maki (beide zufällig):

```
Please enter the population size: 1000000
Do you want to use the Daley or Maki method? (D/M): m
Do you want one person to ALWAYS propagate? (Y/N): n
Please wait! Based on the entered number this can take a while...
nitwits:          13.4%
propagators:      0.0%
suppressors:      86.6%
```

Populationsgröße 1.000.000 – Maki (immer ein Propagator):

```
Please enter the population size: 1000000
Do you want to use the Daley or Maki method? (D/M): M
Do you want one person to ALWAYS propagate? (Y/N): y
Please wait! Based on the entered number this can take a while...
nitwits:          20.3%
propagators:      0.0%
suppressors:      79.7%
```



## Fehlerliste:

1. Bei meiner ersten Implementierung wurde zwar die Anzahl richtig mitgeführt, aber bei den Zuweisungen im Vektor wurde die Zuweisung mit einem Operanden vertauscht (= statt ==). Dadurch waren mehr Propagator im Vektor als die Anzahl festhielt.
2. Mein größtes Problem: Der Zufallsgenerator war unzureichend. Rand() generiert nur Zahlen bis ca. 32000. Somit funktionieren Simulationen mit Populationsgröße über 32000 nicht. Eine neue Random Methode musste gesucht werden.
3. Ich ging davon aus, dass das Statement „Wähle ein zufälliges X aus der Menge der Verbreiter“ der „gesuchte Bug“ des Autors war. Im Endeffekt war es aber die Lösung für das Problem, dass meine Modelle nicht die im Text beschriebenen Theta-Werte erreichten.
4. Ich ließ, bis kurz vor Erstellung dieses Dokuments, die Simulation für Bevölkerungszahlen  $\leq 0$  zu.
5. Der Text wurde anfangs nicht zeilenweise ins Dokument geschrieben, sondern in einem String zwischengespeichert, welcher dann als Ganzes ins File geschrieben wurde. Resultat war eine hohe Arbeitsspeicher-Auslastung.