

Matrix mit Zufallszahlen

Schreiben Sie ein MATLAB-Skript

```
primesub,
```

in dem Sie folgendes durchführen:

1. Erzeugen Sie eine 100×100 Matrix M die ganzzahlige, gleichverteilte Zufallszahlen zwischen 1 und 100 enthält (verwenden Sie `ceil` zum Runden).
2. Speichern Sie den größten in M vorkommenden Wert in der Variablen `max_val`.
3. Erzeugen Sie einen Vektor `v_prim` der alle Primzahlen bis `max_val` enthält (`primes`). Da sieht dann so aus:

2 3 5 7 11 13 17 19 ...

4. Definieren Sie zwei Matrizen M_{p1} und M_{p2} , die gleich M sein sollen.
5. Nehmen sie das erste Element aus `v_prim(1)`. Untersuchen Sie für jede Zahl in M_{p1} , ob `v_prim(1)` ihr Primfaktor ist. (am besten mit der Modulo Funktion `mod`). Falls `v_prim(1)` ihr Primfaktor ist, ersetzen Sie diese Zahl durch `v_prim(1)`. Speichern Sie in `a_prim1(1)` die Anzahl der Zahlen in M_{p1} , die durch `v_prim(1)` ersetzt wurden. Wiederholen Sie diese Schritte für alle Elemente aus `v_prim`.

M_{p1} könnte die folgenden Elemente enthalten :

2, 4, 6, 8, 10, ...	würden ersetzt durch 2.
3, 9, 15, 21, ...	würden ersetzt durch 3.
5, 25, 35, ...	würden ersetzt durch 5.
7, 49, ...	würden ersetzt durch 7.
etc.	

In `a_prim1` könnte dann stehen 5, 4, 3, 2 etc..

6. Ersetzen Sie in M_{p2} jede Zahl hintereinander durch ihre Primfaktoren (beginnend beim kleinsten), bis zum Schluss der größte Primfaktor stehen bleibt. Testen Sie dazu, ob `v_prim(k)` die Elemente in M ohne Rest teilt. Zählen Sie dann für jeden Schritt die Anzahl der Zahlen in M die durch den aktuellen Primfaktor `v_prim(k)` teilbar sind, und speichern Sie das Ergebnis im Vektor `a_prim2(k)`.

In `a_prim2(k)` steht also die Anzahl an Elementen in M die durch den Primfaktor `v_prim(k)` teilbar sind. Hier sind im Gegensatz zum vorherigen Punkt Mehrfachzählungen möglich. Steht in M eine 3 dann trägt dieses Element nur zu `a_prim(2)` bei. Steht in M allerdings eine 6 dann trägt diese zu `a_prim(1)` und `a_prim(2)` bei. Zum Schluss stehen in M_{p2} die größten Primfaktoren der Zahlen in M .

Hinweis:

Mit Hilfe von `mod` können Sie feststellen, ob eine Zahl durch eine andere teilbar ist, und somit ein logisches Feld erzeugen, dass Sie anschließend zur Indizierung und zum Zählen verwenden.

7. Zählen Sie nun noch wie oft der Wert 1 in M vorkommt. Erweitern Sie dann die Vektoren `a_prim1` und `a_prim2` so, dass diese Anzahl das erste Element in diesen Vektoren darstellt.