

## Aufgabenblatt 6 (Zusatzblatt)

Gegeben ist ein eindimensionales Array mit `int`-Elementen aus dem Intervall  $[0, 100)$ . Das Array soll durch Herauslösen einzelner Elemente schrittweise komprimiert werden, bis nur noch die beiden Randelemente übrig sind. Bei jedem Herauslösen ergibt sich ein Gewinn, der für das Herauslösen von  $a[i]$  folgendermaßen berechnet wird: Wir betrachten die Elemente  $a[i-1]$ ,  $a[i]$  und  $a[i+1]$  und lesen deren Werte hintereinander als eine maximal sechsstellige Zahl  $m$ . Der Gewinn beim Herauslösen von  $a[i]$  entspricht dem größten Primteiler von  $m$ .

Der Gesamtgewinn hängt von der Reihenfolge des Herauslösens ab. Beispiel:

a:     27    32    9    28

- 1. Herauslösen von 32  $\rightarrow m = 27329 \rightarrow$  Gewinn = 27329 (da Primzahl), Restarray:  
27   9   28  
2. Herauslösen von 9  $\rightarrow m = 27928 \rightarrow$  Gewinn = 3491, Restarray: 27   28  
 $\rightarrow$  Gesamtgewinn 30820
- 1. Herauslösen von 9  $\rightarrow m = 32928 \rightarrow$  Gewinn = 7, Restarray: 27   32   28  
2. Herauslösen von 32  $\rightarrow m = 273228 \rightarrow$  Gewinn = 22769, Restarray: 27   28  
 $\rightarrow$  Gesamtgewinn 22776

Bearbeiten Sie eine der unten beschriebenen Aufgabenvarianten. Bei allen muss sich Ihr Programm mittels

```
java Prime n seed
```

aufrufen lassen. Dabei bezeichnet  $n$  die Arraylänge und  $seed$  den Initialwert des Zufallszahlengenerators. Initialisieren Sie  $a$  durch

```
Random random = new Random(seed);  
for (int i = 0; i < n; ++i) {  
    a[i] = random.nextInt(100);  
}
```

Addieren Sie die Gewinne im Bereich `long`. Falls es dennoch zu einem Überlauf kommt, ist dieser zu ignorieren, d.h., nach der größten positiven kommt wieder die kleinste negative `long`-Zahl usw.

Bewertet werden **nur korrekte** Lösungen. Falls Sie sich unsicher sind, ob Ihre Lösung einer fortgeschrittenen Variante korrekt ist, geben Sie bitte zusätzlich die Minimal-/Normalvariante mit ab. In diesem Blatt werden ausnahmsweise **keine Teilpunkte** vergeben.

## Minimalvariante

Schreiben Sie ein Java-Programm, welches aus  $a$  solange Elemente herauslöst, bis nur noch die Randelemente übrig sind. Es soll wiederholt immer das Element auf Position 1 entfernt werden. Ausgabe des Programms soll der erzielte Gesamtgewinn sein.

Für korrekte Lösungen dieser Aufgabe gibt es keine Punkte. Jedoch gilt das Blatt als abgegeben, womit es eine Nicht-Abgabe ersetzt.

## Normalvariante

Schreiben Sie ein Java-Programm, welches eine möglichst günstige Reihenfolge für das Herauslösen ermittelt, so dass der Gesamtgewinn maximiert wird. Das Programm soll den erreichten bestmöglichen Gesamtgewinn sowie eine (beliebige) dazu passende Reihenfolge des Herauslösens ausgeben. Beispiel:

```
java Prim 4 23456
a: 27 32 9 28
bestGain = 30820
steps: 1 1
```

Bitte nummerieren Sie die Schritte wie im Beispiel gezeigt, also laut *aktuellem* Arrayindex.

Für korrekte Lösungen dieser Aufgabe gibt es 20 **Zusatzpunkte**. Zusätzlich gilt das Blatt als abgegeben, womit es eine Nicht-Abgabe ersetzt.

## Wettbewerb

Alle korrekten Programme der Normalvariante dürfen am Programmierwettbewerb teilnehmen. Das schnellste Programm wird prämiert! Um am Wettbewerb teilzunehmen, beachten Sie bitte die genaueren Informationen und Regeln für die Teilnahme, welche Anfang Juni auf der Moodle-Seite veröffentlicht werden.