



Department Informatik  
Markus Püschel  
Peter Widmayer  
Thomas Tschager  
Tobias Pröger  
Tomáš Gavenčíak

20. Oktober 2016

## Datenstrukturen & Algorithmen

## Blatt P5

## HS 16

**Abgabe:** Bis Donnerstag, 27. Oktober 2016 um 10 Uhr auf dem Judge (ausschliesslich Quellcode).

### Aufgabe P5.1 *Binäre Suche von Funktionswerten.*

In der Code-Vorlage ist eine Funktion  $f(x)$ , als `int f(int x)` deklariert, gegeben. Die Funktion ist für alle  $x \in \{0, 1, \dots, x_{max}\}$  mit  $x_{max} = 20\,000\,000$  definiert und ist in diesem Bereich monoton steigend, d.h.  $0 \leq i < j \leq x_{max}$  gilt  $f(i) < f(j)$ . Alle Werte sind Ganzzahlen.

Als Eingabe bekommt Ihr Programm  $n \geq 1$  Ganzzahlen  $a_0$  bis  $a_{n-1}$ . Für jedes  $a_i$  sollen Sie  $x_i$  mit  $a_i = f(x_i)$  und  $0 \leq x_i \leq x_{max}$  finden oder ausgeben, dass kein solches  $x_i$  existiert.

**Eingabe** Die Eingabe besteht aus zwei Zeilen. Die erste Zeile enthält lediglich die Ganzzahl  $n$ . Die zweite Zeile enthält  $n$  Ganzzahlen  $a_0$  bis  $a_{n-1}$ , durch Leerzeichen getrennt.

**Ausgabe** Die Ausgabe soll  $n$  Zeilen enthalten, je eine Zeile für jedes  $a_i$ : entweder den Wert von  $x_i$  sodass  $a_i = f(x_i)$  oder den String `NO` (gross geschrieben), falls kein solches  $x_i$  existiert.

**Bonus** Sie erhalten 2 Bonuspunkte wenn Ihr Programm für alle Eingaben funktioniert. Die Laufzeit Ihres Programms soll der Laufzeit der binären Suche für jedes  $a_i$  entsprechen, d.h. die Funktion  $f$  soll nur  $\mathcal{O}(n \log x_{max})$  mal aufgerufen werden. Insbesondere darf  $f$  nicht für alle Werte  $0, \dots, x_{max}$  aufgerufen werden.

Senden Sie Ihr `Main.java` unter folgendem Link ein: [https://judge.inf.ethz.ch/team/websubmit.php?cid=18985&problem=DA\\_P5.1](https://judge.inf.ethz.ch/team/websubmit.php?cid=18985&problem=DA_P5.1). Das Passwort für die Einschreibung ist "quicksort".

### Beispiele

*Eingabe:*

```
6
12 6 13 6 0 2
```

*Ausgabe:*

```
4
3
NO
3
0
NO
```

*Hinweis:*  $f(0) = 0, f(1) = 1, f(2) = 4, f(3) = 6, f(4) = 12, f(5) = 15$ .

**Hinweis** Wir stellen für diese Aufgabe eine Programmvorlage als Eclipse Projektarchiv auf der Vorlesungswebseite zur Verfügung. In der Vorlage wird die Eingabe bereits eingelesen und die Funktion  $f$  definiert. Das Archiv enthält weitere Beispiele – Sie können diese als Eingabe für Ihr Programm verwenden und die Ausgabe überprüfen.

Wir empfehlen dieses Problem mit binärer Suche für  $x$  im Bereich  $0 \dots x_{max}$  für jedes gegebene  $a$  zu lösen. Stellen Sie sich vor, dass jeder Wert von  $f$  für  $0 \dots x_{max}$  in einem Array mit  $x_{max} + 1$  Elementen gespeichert ist, Sie aber nur diejenigen Werte von  $f$  berechnen, die sie tatsächlich brauchen. Sie können sowohl die einfache binäre Suche mit einer While-Schleife als auch die rekursive binäre Suche verwenden.

Ausserdem sollten Sie nicht versuchen, die Funktion  $f$  zu analysieren und die Inverse direkt zu berechnen – betrachten Sie  $f$  als Blackbox. (Natürlich können Sie es versuchen.)