Inhaltsverzeichnis

[Sequences 2](#_Toc23673774)

[Lösungsidee 2](#_Toc23673775)

[Code 2](#_Toc23673776)

[Testfälle 2](#_Toc23673777)

[Teambuilding 2](#_Toc23673778)

[Lösungsidee 2](#_Toc23673779)

[Code 2](#_Toc23673780)

[Testfälle 2](#_Toc23673781)

# Sequences

## Lösungsidee

### Longest increasing run

Den längsten monoton steigenden Lauf einer gegebenen Folge kann in linearer Zeit mit einem Schleifendurchlauf ermittelt werden. Dafür muss lediglich eine count- und eine previous\_count Variable mitgeführt werden. Die Schleife durchwandert alle Elemente im Feld. Startindex ist 1, da immer auf das Element davor zugegriffen wird. In der Schleife wird geprüft, ob das vorherige Element kleiner ist als das derzeitige. Wenn dies der Fall ist, so wird die count Variable um eins erhöht. Sollt dies nicht der Fall sein, so wird geprüft, ob count > previous\_count ist (TRUE 🡪 previous count = count || FALSE 🡪 count = 1).

### Longest increasing subsequence

Als default kann für das erste Element als sub-sequence length immer 1 angenommen werden. Das erste Element besitzt außerdem keinen Vorgänger, daher kann hier der predecessor auf INT\_MIN gesetzt werden. Grundsätzlich muss über alle Elemente der Sequence iteriert werden. Für jedes Element werden alle Elemente davor betrachtet und gecheckt, welches kleiner als das jetzige ist und die größte sub-sequence length hat. Im Anschluss wird max sequence length an der stelle i im Feld L gespeichert. Die Position vom predecessor wird ebenfalls in einem Feld abgelegt. Ist die Sequence fertig durchlaufen, so kann die Tabelle wie in der Angabe ausgegeben werden. Die sub-sequences können mit Hilfe des Feldes P rekonstruiert werden.

## Code

|  |
| --- |
|  |

## Testfälle

# Teambuilding

## Lösungsidee

## Code

|  |
| --- |
|  |

## Testfälle