

Aufgabe 1: L^AT_EX-Dokument

Team-ID: 00000

Team-Name: Name

Bearbeiter/-innen dieser Aufgabe:
Vor- und Nachnamen

9. Oktober 2018

Inhaltsverzeichnis

1	Lösungsidee	1
1.1	Unterüberschrift	1
1.1.1	Unter-Unterüberschrift	1
1.2	Mathematik	1
1.3	Algorithmen	2
2	Umsetzung	3
3	Beispiele	3
4	Quellcode	3

Anleitung: Trage oben in den Zeilen 8 bis 11 die Aufgabennummer, die Team-ID, den Team-Namen und alle Bearbeiter/-innen dieser Aufgabe mit Vor- und Nachnamen ein. Vergiss nicht, auch den Aufgabennamen anzupassen (statt „L^AT_EX-Dokument“)!

Dann kannst du dieses Dokument mit deiner L^AT_EX-Umgebung übersetzen.

Die Texte, die hier bereits stehen, geben ein paar Hinweise zur Einsendung. Du solltest sie aber in deiner Einsendung wieder entfernen!

1 Lösungsidee

Die Idee der Lösung sollte hieraus vollkommen ersichtlich werden, ohne dass auf die eigentliche Implementierung Bezug genommen wird.

Der Text wird am besten mit Unterüberschriften strukturiert:

1.1 Unterüberschrift

1.1.1 Unter-Unterüberschrift

Benannter Absatz Lorem ipsum.

1.2 Mathematik

$$a_0 := 2 + 3 \tag{1}$$

$$= 3^2 - 2^2 \tag{2}$$

$$1/4$$

$$\sum_{i=0}^3 i = \prod_{i=1}^3 i = \frac{12}{2} = \sqrt[3]{216} \quad (3)$$

$$f(x) \propto x^2 \Leftrightarrow \exists r \in \mathbb{R} : \forall x : f(x) = r * x^2 \Rightarrow x \in \mathcal{O}(x^2)$$

Mathematische Formeln erhalten normalerweise eine Nummer, die man referenzieren kann. In Gleichung (3) zum Beispiel ist jeder Term gleich 6. Mathematische Formeln können auch im Fließtext gesetzt werden: $\sum_{i=0}^3 i = \prod_{i=1}^3 i = \frac{12}{2} = \sqrt[3]{216}$.

1.3 Algorithmen

```

if  $i \geq i_{\max}$  then
   $a \leftarrow -1$ 
else if  $i \leq i_{\min}$  then
   $a \leftarrow 1$ 
else
   $a \leftarrow 0$ 
end if
for  $i$  from 1 to 10 do
   $a \leftarrow a + i$ 
end for
for all  $i$  in Liste do
   $a \leftarrow a + i$ 
end for
while  $a > 0$  do
   $a \leftarrow a - 1$ 
end while
repeat
   $a \leftarrow a - 1$ 
until  $a < 0$ 
loop
   $i \leftarrow \text{INCREMENT}(i)$ 
end loop
function INCREMENT( $a$ )
   $a \leftarrow a + 1$ 
  return  $a$ 
end function

```

1.4 Bilder

Bilder werden von L^AT_EX nicht direkt da angezeigt, wo sie eingebunden wurden, sondern an passenden Stellen in der Nähe. Das ist so beabsichtigt. Man kann aber Bilder referenzieren. Auf Abb. 1 sieht man irgendwas mit Informatik, auf Abb. 2 auch.

Man kann Bilder fest positionieren, dann können sie aber keine Bildunterschrift haben und man kann sie nicht referenzieren. Zum Beispiel diese Veranschaulichung einer Queue (Warteschlange):

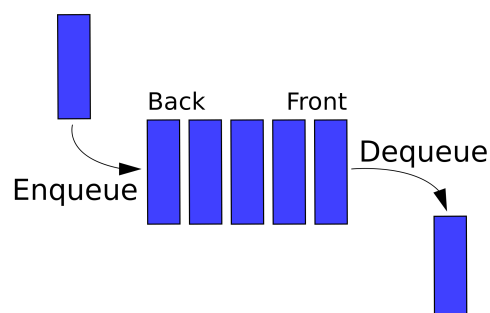




Abbildung 1: Ein Bild mit irgendwas mit Informatik (Lizenz: CC0)

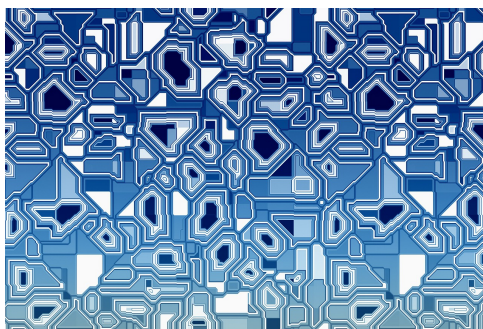


Abbildung 2: Ein anderes Bild mit irgendwas mit Informatik (Lizenz: CC0)

2 Umsetzung

Hier wird kurz erläutert, wie die Lösungsidee im Programm tatsächlich umgesetzt wurde. Hier können auch Implementierungsdetails erwähnt werden.

3 Beispiele

Genügend Beispiele einbinden! Die Beispiele von der BwInf-Webseite sollten hier diskutiert werden, aber auch eigene Beispiele sind sehr gut – besonders wenn sie Spezialfälle abdecken. Aber bitte nicht 30 Seiten Programmausgabe hier einfügen!

4 Quellcode

Unwichtige Teile des Programms sollen hier nicht abgedruckt werden. Dieser Teil sollte nicht mehr als 2–3 Seiten umfassen, maximal 10.

```
1 for i in range(100):  
  if i % 2 == 0:  
3 pass  
  else:  
5 print(i)
```

Kleines Python-Programm

```
1 #include <iostream>  
  
3 int main(int argc, char** argv) {  
  std::cout << "Hallo";  
5 return 0;  
}
```

Kleines C++-Programm

Hier noch ein kurzes Pascal-Snippet ohne Rahmen:

```
for i := maxint to 0 do
2 begin
  a := a + i
4 end;
Write('Case_ insensitive_');
6 Write('Pascal_keywords_');
```

Das folgende Programm wird aus einer Datei geladen:

```
"use_strict";
2 /* Beispiel-JavaScript-Datei, die in das LaTeX-Dokument eingebettet
   * wird. */
4 console.log("Hallo_Welt");
```

Großes JS-Programm