

## Aufgabenblatt 7, Abgabetermin 29.6.2020

---

Lösen Sie drei Aufgaben. Sie können maximal 15 Punkte erreichen.

---

### Aufgabe 42: Prüfziffern in der IBAN

5 Punkte

Die IBAN (Internationale Bankkontonummer) besteht aus 2 Buchstaben (Länderkürzel) gefolgt von 2 Ziffern (Prüfziffern) und der Kontoidentifikation aus bis zu 30 Großbuchstaben und Ziffern. Bei einer deutschen IBAN – z.B. DE87123456781234567890 – besteht die Kontoidentifikation aus der 8-stelligen Bankleitzahl und der 10-stelligen Kontonummer.

Die Prüfziffern werden aus einer Zahl berechnet, die wie folgt entsteht: man nimmt zuerst die Kontoidentifikation (im Beispiel 123456781234567890) und hängt daran das Länderkürzel (im Beispiel DE) und zwei Nullen. Alle Großbuchstaben 'A', 'B', 'C', 'D', 'E', ..., 'Z' werden durch Zahlen 10, 11, 12, 13, 14, ..., 35 ersetzt. Im Beispiel haben wir dann die Zahl  $i=123456781234567890131400$  erhalten. Die Prüfziffern werden als  $98 - i\%97$  berechnet. In dem Beispiel hat  $i\%97$  den Wert 11. Die Prüfziffern sind also 87. (Falls die Prüfziffern nur einstellig sind, werden sie mit einer führenden 0 geschrieben.)

Schreiben Sie eine Funktion, die mit Länderkürzel und Kontoidentifikation aufgerufen wird und die IBAN zurückgibt, und eine Funktion, die mit einer (möglichen) IBAN aufgerufen wird und zurückgibt (`True` oder `False`), ob die IBAN korrekt ist.

Schreiben Sie ein Programm, das diese beiden Funktionen testet.

(Hinweis: die Python-Funktion `ord(x)` liefert eine Zahlencodierung des Zeichens `x`. `ord('A')` gibt 65 zurück, `ord('B')` gibt 66 zurück usw. bis `ord('Z')`, das 90 zurückgibt.)

### Aufgabe 43: Array aus Strings in Array aus Zahlen umwandeln

5 Punkte

Schreiben Sie eine Funktion, die mit einem Array aus Strings aufgerufen wird und ein Array aus `int`- und `float`-Werten zurückgibt. Sie können davon ausgehen, dass die Strings tatsächlich Zahlen darstellen. Wenn die Darstellung einen '.' enthält, soll der String in einen `float`-Wert umgewandelt werden. Sonst soll er in einen `int`-Wert umgewandelt werden.

Schreiben Sie ein Programm zum Testen der Funktion, das das Array `sys.argv[1:]` der Eingabewerte auf der Kommandozeile als Array aus Zahlen ausgibt.

### Aufgabe 44: Arrays normalisieren

5 Punkte

Arrays aus `float`-Zahlen sollen so verändert werden, dass alle Zahlen im Bereich  $-1 \dots 1$  liegen, aber die Größenverhältnisse zwischen den Zahlen erhalten bleiben. Dazu werden alle Einträge im Array durch den betragsmäßig größten Wert im Array geteilt. Zum Beispiel wird das Array `[1, -1.6, 4, 2]` normalisiert zu `[0.25, -0.4, 1, 0.5]`, und das Array `[1, 1.6, -4, 2]` wird normalisiert zu `[0.25, 0.4, -1, 0.5]`.

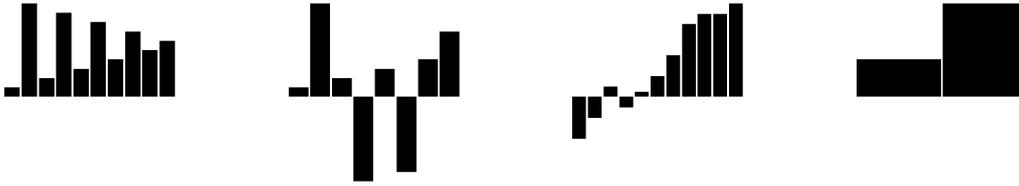
Schreiben Sie zwei Funktionen, die ein Array als Argument erhalten und es normalisieren. Die erste Funktion soll das Argument-Array ändern und keinen Rückgabewert haben. Die zweite Funktion soll das Argument-Array unverändert lassen und ein neues Array zurückgeben.

Schreiben Sie ein Programm, das die Elemente des Arrays von der Kommandozeile liest und die beiden Funktionen damit testet

### Aufgabe 45: Säulendiagramm malen

5 Punkte

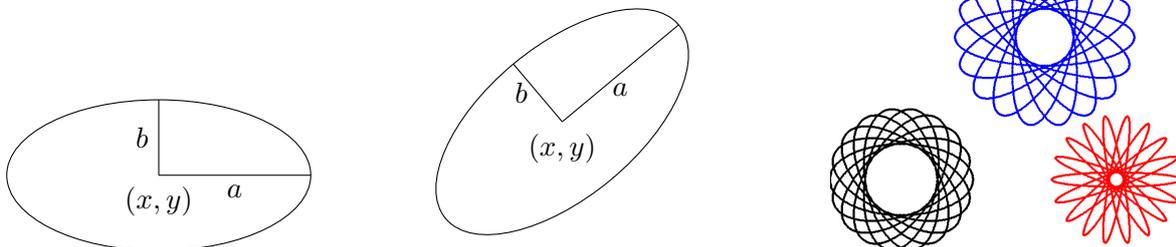
Schreiben Sie eine Funktion, die ein Array mit Zahlen als Argument erhält und diese Werte als Säulendiagramm auf die Leinwand von `std::draw` malt. Sie können dabei Funktionen aus den vorigen Aufgaben benutzen.



Schreiben Sie ein Programm, das die Funktion testet, indem eingegebene Werte als Säulendiagramm dargestellt werden.

### Aufgabe 46: Ellipsen malen

5 Punkte



eine Ellipse

eine Ellipse, die  $40^\circ$  um ihren  
Mittelpunkt gedreht wurde

viele bunte Ellipsen

Eine Ellipse wird durch ihren Mittelpunkt  $(x, y)$  und die beiden Längen  $a$  und  $b$  beschrieben (siehe Bild). Sie sollen gedrehte Ellipsen malen. Dazu kann man sich folgendes überlegen.

Eine Ellipse mit dem Mittelpunkt  $(0, 0)$  kann man malen, indem man für alle Winkel  $t$  den Punkt  $(a \cdot \cos t, b \cdot \sin t)$  (die „Ellipsenpunkte“) malt.

Ein geometrisches Objekt, das aus Punkten  $(p, q)$  besteht, kann man  $r^\circ$  um den Punkt  $(0, 0)$  drehen, indem man jeden Punkt  $(p, q)$  nach  $(p \cdot \cos r - q \cdot \sin r, p \cdot \sin r + q \cdot \cos r)$  verschiebt.

Ein geometrisches Objekt mit Mittelpunkt  $(0, 0)$ , das aus Punkten  $(p, q)$  besteht, kann man zum Mittelpunkt  $(x, y)$  verschieben, indem man jeden Punkt  $(p, q)$  nach  $(p + x, q + y)$  verschiebt.

Schreiben Sie drei („einzeilige“) Funktionen zum (1) Berechnen des Ellipsenpunktes für Parameter  $a, b, t$  bei Mittelpunkt  $(0, 0)$ , (2) Drehen eines Punktes  $(p, q)$  in einem Objekt mit Mittelpunkt  $(0, 0)$  um  $r^\circ$ , (3) Verschieben eines Punktes  $(p, q)$  in einem Objekt mit Mittelpunkt  $(0, 0)$  zu einem anderen Mittelpunkt  $(x, y)$ . Jede dieser Funktionen gibt einen Punkt bzw. dessen Koordinaten zurück.

Schreiben Sie eine Funktion mit den Parametern  $x, y, a, b, r$ , die eine Ellipse mit Mittelpunkt  $(x, y)$  und Längen  $a, b$  malt, die  $r^\circ$  gedreht ist. Lassen Sie die Funktion jeden Ellipsenpunkt zuerst berechnen, dann drehen, dann verschieben und schließlich malen. Benutzen Sie dabei die obigen 3 Funktionen.

Testen Sie Ihr Programm, indem Sie ein ähnliches Bild wie die bunten Ellipsen oben erzeugen.