
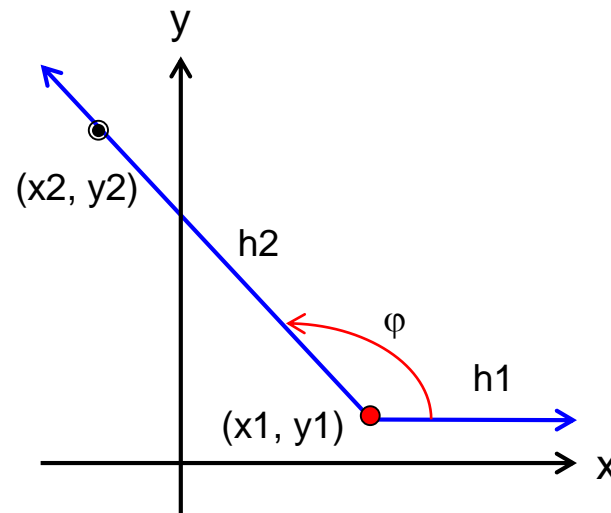


- Im Praktikum erstellen wir die autonome Steuerung (Autopilot) einer Lieferdrohne. Dazu gehören u.a. 
- autonome Steuerung mit Steuerbefehlen
- Routenplanung: Es sollen mehrere Ziele angesteuert werden
- Protokollierung der Flugstrecke
- Die Drohne wird simuliert. Lesen Sie zum Verständnis der Simulation bitte das Ufo-Handbuch.
- Bevor Sie beginnen, sollten Sie die Entwicklungsumgebung auf Ihrem Rechner installiert haben (siehe Anleitung). Oder Sie verwenden die Rechner in den PC-Pools. Auf jeden Fall müssen Sie sich in die Verwendung der Entwicklungsumgebung einarbeiten.
- Im **ersten Teil des Praktikums** sollen einige Skripte programmiert werden, die später für das Ufo nützlich sind.

Erstellen Sie Ablaufdiagramme und Python-Skripte für folgende Aufgabenstellungen.

1. Eingabe: Fließkommazahlen x_1, y_1, x_2, y_2 : zwei Punkte im kartesischen Koordinatensystem

Ausgabe: Fließkommazahl : Winkel φ in Grad, zwischen der Halbgeraden h_1 und h_2 ,
(siehe Abbildung), wobei h_1 parallel zur x -Achse ist und $0^\circ \leq \varphi < 360^\circ$





2. Eingabe: Fließkommazahl x
Ganze Zahl n

Ausgabe: Fließkommazahl : $\cos(x)$ berechnet durch die Näherung $\cos(x) = 1 - x^2/2! + x^4/4! - x^6/6! + \dots$
Fließkommazahl : Fehler

Es sollen n Summanden bei der Näherung berücksichtigt werden. Außerdem soll der Fehler bestimmt werden, indem die Differenz von $\text{math.cos}(x)$ und dem Näherungswert berechnet wird. Die Ausgabe soll wie im folgenden Beispiel aussehen:

$\cos(-3.14) = -0.9999987316527268$

Fehler: $-7.481271158127356e-11$

Dieses Skript wird für das Ufo nicht benötigt.



3. Um den Flug des Ufos auf der Konsole nachvollziehen zu können, sollen Flugdaten in Tabellenform ausgegeben werden. Bei Fließkommazahlen soll dabei nur eine Nachkommastelle erscheinen. Schreiben Sie ein Skript, das nebenstehende Konsolausgabe erzeugt (kein Ablaufdiagramm erforderlich). Die t- und z-Spalte sollen jeweils fünf Zeichen breit sein, die x- und y-Spalte sechs Zeichen breit. Die Werte werden ausgehend von 0.0 von Zeile zu Zeile jeweils um folgenden Wert erhöht: $\Delta t = 0.5$, $\Delta x = 5.25$, $\Delta y = -0.18$, Δz soll eingelesen werden. Bei nebenstehender Beispielausgabe war $\Delta z = 0.1$.

Die Formatierung kann mit der Funktion `format` erreicht werden. Im Kapitel 12.4 in (Klein 2018) finden Sie Beispiele dazu. Beispiel:

```
print("Artikel:~{0:5d},~Preis:~{1:8.2f}".format(453, 59.058))
```

ergibt Artikel:~~~453,~Preis:~~~~59.06

(Die Zeichen ~ sollen lediglich die Leerzeichen sichtbar machen.)

t		x	y	z
0.0	s: [0.0	-0.0	0.0]
0.5	s: [5.2	-0.2	0.1]
1.0	s: [10.5	-0.4	0.2]
1.5	s: [15.8	-0.5	0.3]
2.0	s: [21.0	-0.7	0.4]
2.5	s: [26.2	-0.9	0.5]
3.0	s: [31.5	-1.1	0.6]
3.5	s: [36.8	-1.3	0.7]
4.0	s: [42.0	-1.4	0.8]
4.5	s: [47.2	-1.6	0.9]
5.0	s: [52.5	-1.8	1.0]
5.5	s: [57.8	-2.0	1.1]
6.0	s: [63.0	-2.2	1.2]
6.5	s: [68.2	-2.3	1.3]
7.0	s: [73.5	-2.5	1.4]
7.5	s: [78.8	-2.7	1.5]
8.0	s: [84.0	-2.9	1.6]
8.5	s: [89.2	-3.1	1.7]
9.0	s: [94.5	-3.2	1.8]
9.5	s: [99.8	-3.4	1.9]
10.0	s: [105.0	-3.6	2.0]

■ Abgabe:

- Sie können die Ablaufdiagramme zu Aufgabe 1 und 2 in PowerPoint, Word oder einem beliebigen anderem Zeichenprogramm erstellen. Auf Moodle finden Sie eine PowerPoint-Datei als Muster. Exportieren bzw. konvertieren Sie die Diagramme nach der Erstellung in eine pdf-Datei.
- Die drei Skripte schreiben Sie bitte in eine py-Datei. Testen Sie die Skripte ausgiebig mit verschiedenen Werten. Wenn alle Tests erfolgreich waren, verpacken Sie die pdf-Datei und die py-Datei in eine zip-Datei. Laden Sie die zip-Datei anschließend in Moodle hoch.
- Bitte laden Sie die zip-Datei rechtzeitig hoch. Die zip-Datei enthält genau zwei Dateien.

- Checkliste: Überprüfen Sie vor dem Hochladen der Abgabe, ob Folgendes erfüllt ist:
 - Die Abgabe ist eine zip-Datei, die eine pdf-Datei und eine py-Datei enthält.
 - Die pdf-Datei enthält zwei Ablaufdiagramme.
 - Die py-Datei enthält drei Python-Skripte.
 - Das Skript zur Berechnung des Winkels (Aufgabe 1) liest die benötigten Eingabewerte mit input von der Konsole, berechnet den Winkel und gibt das Ergebnis mit print auf die Konsole aus.
 - Das Skript zur Berechnung des Kosinus (Aufgabe 2) liest die benötigten Eingabewerte mit input von der Konsole, berechnet den Kosinus und gibt das Ergebnis mit print auf die Konsole aus.
 - Das Skript zur Ausgabe der Flugdatentabelle (Aufgabe 3) liest den benötigten Eingabewert mit input von der Konsole und gibt die Tabelle mit print auf die Konsole aus.
 - Alle drei Skripte haben keine Syntaxfehler.
 - Alle drei Skripte sind mit verschiedenen Eingabewerten getestet und fehlerfrei.