

☐ **Gr. 1, DI (FH) G. Horn-Völlenkle, MSc** **Name** _____ **Aufwand in h** _____☐ **Gr. 2, S. Schöberl, MSc****Punkte** _____ **Tutor*in / Übungsleiter*in** ____ / ____

1. Algorithmus: Einnahmen/Ausgaben-Rechnung (4 + 4 Punkte)

Entwickeln Sie einen Algorithmus, der eine Folge ganzer Zahlen einliest und die Summe der positiven Zahlen (*Einnahmen*) sowie die Summe der negativen Zahlen (*Ausgaben*) bildet. Wird der Wert 0 gelesen, soll der Algorithmus das Einlesen beenden und die beiden Summen sowie eine Gesamtsumme (= Einnahmen – Ausgaben) ausgeben. Stellen Sie den Algorithmus mittels (a) *Pseudocode* und (b) *Ablaufdiagramm* dar.

Beispiele:

1. Eingabe:	10 20 -30 40 -50 60 70 80 -90 0
Ausgabe:	Einnahmen: 280
	Ausgaben: -170
	Gesamt: 110
2. Eingabe:	0
Ausgabe:	Einnahmen: 0
	Ausgaben: 0
	Gesamt: 0

2. Pascal-Programm: Einnahmen/Ausgaben-Rechnung (6 Punkte)

Implementieren Sie den Algorithmus aus Aufgabe 1 in Pascal. Testen Sie Ihr Programm ausführlich und geben Sie aussagekräftige Testfälle ab.

3. Pascal-Programm: tägliche Zeiterfassung (6 Punkte)

Entwickeln Sie ein Pascal-Programm, welches die tägliche Arbeitszeit in Form ganzzahliger Werte für Stunden und Minuten einliest und aus ggf. geleisteten Überstunden den Anspruch auf Zeitausgleich ermittelt und ausgibt. Bezüglich der täglichen Arbeitszeit gelten (nur hier!) folgende Regeln

- Die ersten acht Stunden sind Normalarbeitszeit, es fallen keine Überstunden an.
- Die 9. und 10. Stunde kann als Zeitausgleich in Anspruch genommen werden.
- Die 11. und 12. Stunde kann mit Faktor 1,5 als Zeitausgleich in Anspruch genommen werden.
- Täglich dürfen höchstens 12 Stunden gearbeitet werden.

Beispiele:

1. Eingabe:	8 30
Ausgabe:	Anspruch auf Zeitausgleich: 0.50 Stunden
2. Eingabe:	11 00
Ausgabe:	Anspruch auf Zeitausgleich: 3.50 Stunden
3. Eingabe:	13 00
Ausgabe:	Tägliche Hoechst Arbeitszeit ueberschritten

4. Darstellungsformen

(4 Punkte)

Diskutieren Sie die Vor- und Nachteile der Darstellungsformen für Algorithmen, die Sie in Aufgabe 1 und 2 verwendet haben.

Hinweise:

1. Lesen Sie die organisatorischen Hinweise im Moodle-Kurs.
2. Geben Sie für alle Ihre Lösungen immer eine „Lösungsidee“ an.
3. Dokumentieren und kommentieren Sie Ihre Algorithmen.
4. Bei Programmen: Geben Sie immer auch Testfälle ab, an denen man erkennen kann, dass Ihr Programm funktioniert, und dass es auch in Fehlersituation entsprechend reagiert.