

1. (3 Punkte) Schreibe ein Python-Programm, das die folgenden Anweisungen ausführt:

Lies in die Variable **note** eine Zahl (1-6) ein. Setze die Variable **ergebnis** auf 'bestanden', falls die Note mindestens 4 ist. Setze **ergebnis** auf 'nicht bestanden', falls die Note schlechter als 4 ist.

Lösung:

```
note = int(input("Bitte eine Note (1-6) eingeben: "))
if note <= 4:
    ergebnis = "bestanden"
else:
    ergebnis = "nicht bestanden"
```

2. (4 Punkte) Schreibe ein Python-Programm, das die folgenden Anweisungen ausführt:

Lies eine ganze Zahl x ein.

Gib aus:

'positiv', wenn x größer als 0 ist,
 'null', wenn x gleich 0 ist,
 'negativ', wenn x kleiner als 0 ist.

Lösung:

```
x = int(input("Bitte eine ganze Zahl eingeben: "))

if x > 0:
    print("positiv")
elif x == 0:
    print("null")
else:
    print("negativ")
```

3. (4 Punkte) Gib die Wahrheitstafel für den Ausdruck **(A or B) and not A** für alle vier Kombinationen von A und B (jeweils 0 für **False** und 1 für **True**) an.

Lösung:

A	B	(A or B)	not A	(A or B) and not A
0	0	0	1	0
0	1	1	1	1
1	0	1	0	0
1	1	1	0	0

4. (6 Punkte) Gib die Wahrheitstafel für den Ausdruck **(A and not B) or C** für alle acht Kombinationen von A, B und C (jeweils 0 für **False** und 1 für **True**) an.

Lösung:

A	B	C	not B	(A and not B)	(A and not B) or C
0	0	0	1	0	0
0	0	1	1	0	1
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	0	0	0
1	1	1	0	0	1

5. (2 Punkte) In x sei eine ganze Zahl gespeichert. Schreibe einen Python-Ausdruck, der genau dann wahr ist, wenn x größer als 10 ist und durch 5 teilbar ist.

Lösung:

```
x > 10 and x % 5 == 0
```

6. (2 Punkte) In x sei eine ganze Zahl gespeichert. Schreibe einen Python-Ausdruck, der genau dann wahr ist, wenn x zwischen 110 und 120 (einschließlich) liegt oder durch 7 teilbar ist.

Lösung:

```
(110 <= x and x <= 120) or x % 7 == 0
```

7. (3 Punkte) Schreibe mit einer while-Schleife ein Python-Programm, das die Zahlen von 5 bis 12 ausgibt.

Lösung:

```
x = 5
while x <= 12:
    print(x)
    x += 1
```

8. (2 Punkte) Was wird ausgegeben?

```
x = 17
print(x)
while x >= 10:
    x -= 3
print(x)
```

Lösung:

```
17
8
```

9. (3 Punkte) Schreibe mit einer for-Schleife ein Python-Programm, das die Zahlen von 5 bis 12 ausgibt.

Lösung:

```
for x in range(5, 13):
    print(x)
```

10. (3 Punkte) Schreibe mit einer for-Schleife ein Python-Programm, das die Zahlen von 11 bis 7 ausgibt.

Lösung:

```
for x in range(11, 6, -1):
    print(x)
```

11. (3 Punkte) Was wird ausgegeben?

```
i = 2
for k in range(i):
    print(i+k)
```

Lösung:

2
3

12. (2 Punkte) Was wird ausgegeben?

```
i = 5
for k in range(3):
    print(i)
```

Lösung:

5
5
5

13. (4 Punkte) Schreibe ein Python-Programm, das mit einer for-Schleife die Zahlen von 1 bis 30 durchläuft. Überspringe dabei alle Zahlen, die durch 5 teilbar sind (mit `continue`). Breche die Schleife ab (mit `break`), sobald eine Zahl größer als 20 ist. Gib alle nicht übersprungenen Zahlen aus.

Lösung:

```
for i in range(1, 31):
    if i % 5 == 0:
        continue
    if i > 20:
        break
    print(i)
```