



### Programmierung 3: Was wäre, wenn ...

Bei allen Programmieraufgaben muss eine angemessene GUI erstellt werden, über die die Eingabe und Ausgabe erfolgen kann.  
Verwenden Sie Labels, Textfelder und Buttons.

1 Bestimmen Sie die Werte der Variablen  $a$ ,  $b$  und  $c$  am Ende des jeweiligen Programms.

(a)

```
1  int c;
2  int a = 4;
3  int b = 5;
4  if(a<b){
5      c=b-a;
6  }else{
7      c=a+b;
8  }
```

(b)

```
1  int c;
2  int a = 5;
3  int b = 3;
4  if(a >= b+a){
5      a = a+1;
6      c = 2*a+2;
7      if(c == 13){
8          c = c+1;
9      }else{
10         c = c-2;
11     }
12 }else{
13     b = b*3+1;
14     c = a+b;
15     if(c == 15){
16         c = c-7;
17     }else{
18         c = c-1;
19     }
20 }
21 c = c+a;
```

2 Implementieren Sie das Programm `Gruss`: Der Benutzer gibt seinen Namen ein. Anschließend gibt das Programm aus: »Hallo XY«, gefolgt von », das ist ein schöner Name!«, wenn der Benutzer *Ihren* Namen eingegeben hat und », naja, es gibt bessere Namen...«, wenn er einen anderen Namen eingegeben hat.

3 Implementieren Sie das Programm `Sortieren`: Der Benutzer gibt zwei ganze Zahlen ein und das Programm gibt diese aufsteigend sortiert (d.h. erst die kleinere, dann die größere Zahl) aus.

4 Implementieren Sie das Programm `DreiButtons`: Wenn der erste Button geklickt wird, wird 1 ausgegeben, wenn der zweite geklickt wird, wird 2 ausgegeben und wenn der dritte geklickt wird, wird 3 ausgegeben.

5 Implementieren Sie das Programm `Taschenrechner`: Der Benutzer gibt zwei Kommazahlen  $a$  und  $b$  ein und klickt anschließend auf einen der Knöpfe »Plus«, »Minus«, »Mal«, »Geteilt« oder »Potenz«. Das Programm berechnet anschließend  $a + b$  bzw.  $a - b$  bzw.  $a \cdot b$  bzw.  $a : b$  bzw.  $a^b$  und zeigt das Ergebnis an.



- 6 Implementieren Sie das Programm `PQFormel`: Der Benutzer gibt zwei Kommazahlen  $p$  und  $q$  ein und klickt anschließend auf einen Button. Das Programm berechnet anschließend die Lösungen der quadratischen Gleichung

$$x^2 + px + q = 0$$

mit Hilfe der  $p$ - $q$ -Formel

$$x_{1/2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q}$$

und gibt die Lösungen aus. Dabei soll das Programm anzeigen, wenn es keine Lösung gibt und es soll nur einen Wert anzeigen, wenn es genau eine Lösung gibt.

Zur Kontrolle:

- $p = -4, q = 3.5: x_1 \approx 2,707, x_2 \approx 1,293$
- $p = -4, q = 4: x_1 = 2$
- $p = -4, q = 5: \text{keine Lösung}$

- 7 Implementieren Sie das Programm `Zaehlen`: Der Zähler beginnt bei 0. Wird der Button »+« geklickt, so wird der Zähler um eins erhöht, wird der Button »-« geklickt, so wird der Zähler um eins verringert, nicht aber unter 0.

- 8 »Zahlenraten« ist ein beliebtes Rate-Spiel mit zwei Spielern. Der erste Spieler denkt sich heimlich eine Zahl zwischen 1 und 10 aus. Der zweite Spieler versucht daraufhin, die geheime Zahl zu raten. Nach jedem Rateversuch verrät der erste Spieler, ob die geratene Zahl richtig ist, bzw. ob sie zu groß oder zu klein ist.

Das Spiel »Zahlenraten« soll als Computerspiel umgesetzt werden. Dabei soll der Computer die Rolle des ersten Spielers übernehmen. Der Benutzer muss also die geheime Zahl des Computers erraten.

(a) Zeichnen Sie eine geeignete GUI für das Spiel. Implementieren Sie diese GUI.

(b) Implementieren Sie das Spiel unter Beachtung der folgenden Hinweise:

- Der Befehl `Blox.getRandomInt(min, max)` erzeugt eine Zufallszahl zwischen `min` und `max`. Sie finden den entsprechenden Block unter »Mathematik«.
- Erzeugen Sie die geheime Zahl zwischen 1 und 10 direkt bei Programmstart.
- Verwenden Sie Verzweigungen in der `onAction`-Methode, um die richtige Anzeige zu generieren.



9 Eine ganz einfache Variante des Spiels »Mensch, ärgere dich nicht« besteht darin, dass zwei Spieler gegeneinander spielen. Jeder Spieler startet an Position 0 und würfelt, wenn er am Zug ist. Anschließend darf er um die gewürfelte Zahl weiterziehen (d.h., seine Position ändert sich um die gewürfelte Zahl). Wenn er dieselbe Position erreicht wie der andere Spieler, so fällt dieser zurück auf Position 0. Es gewinnt der Spieler, der zuerst exakt die Position 25 erreicht.

Das Spiel soll so implementiert werden, dass zwei menschliche Spieler gegeneinander spielen können.

(a) Implementieren Sie die GUI für das Spiel aus Material 1. Es reicht, dass die aktuelle Position der Spieler als Zahlenwert angezeigt wird. Ansonsten muss erkennbar sein, welcher Spieler gerade am Zug ist. Des Weiteren wird nur ein Button benötigt, nämlich »Würfeln«.

(b) Beschreiben Sie die Aufgaben, die die einzelnen Elemente Ihrer GUI übernehmen sollen.

(c) Implementieren Sie das Spiel unter Beachtung der folgenden Hinweise:

- Sie benötigen zwei Variablen, um die jeweilige Position der Spieler zu speichern.
- Sie benötigen eine weitere Variable, um zu speichern, welcher Spieler gerade am Zug ist.
- Ändern Sie nach jedem Klick auf »Würfeln« die Anzeige des Spielers, der gerade am Zug ist.

Mensch ärgere dich nicht!	
Position Spieler 1:	7
Position Spieler 2:	11
Gewürfelte Zahl:	5
Würfeln!	
Spieler 1 ist am Zug. Klicke auf Würfeln!	

**Material 1** Grafische Oberfläche für Mensch-ärgere-dich-nicht.