

C/C++ Übungsblatt 3 (Block 1)

Prof. Dr. Klaus Obermayer und Mitarbeiter

Kontrollfluss: Bedingte Ausführung und Schleifen

| | |
|---------------|------------|
| Verfügbar ab: | 23.11.2020 |
| Abgabe bis: | 30.11.2020 |

Aufgabe 1: Notenvergabe

3 Punkte

Betrachten Sie das folgende Punktesystem zur Notenvergabe:

| Punkte | Note |
|--------|------|
| 36-40 | 1 |
| 31-35 | 2 |
| 26-30 | 3 |
| 21-25 | 4 |
| 0-20 | 5 |

Schreiben Sie ein C Programm `Notenvergabe.c`. Ihr Programm soll in der `main`-Methode eine ganze Zahl als Eingabe von der Konsole bekommen und anschließend die dazugehörige Note ausgeben.

Zum Einlesen der Punktzahl verwenden sie dann bitte folgenden Code.

```
1 int punkte;  
2 scanf("%d", &punkte);  
3  
4 //Hier Ihr Code
```

Für den Fall, dass eine Punktzahl außerhalb des oben definierten Bereichs eingegeben wird, soll eine Fehlermeldung ausgegeben werden.

Hinweis: Benutzen Sie die in C verfügbaren Vergleichs-Operatoren und verwenden Sie verschachtelte bedingte Anweisungen oder Dijkstra-if's, um alle obigen Fälle abzudecken. Der Lesbarkeit halber können Sie unnötige geschweifte Klammern vermeiden.

Aufgabe 2: For- und while-Schleifen

2 Punkte

Implementieren Sie ein Programm `Counter.c`, dass die Zahlen 0 bis 10 und 10 bis 0 auf der Konsole ausgibt. Verwenden Sie zuerst eine `for`-Schleife, um von 0 zu 10 hochzuzählen. Nutzen Sie danach eine `while`-Schleife, um von 10 zu 0 herunterzuzählen. Verwenden Sie zum Hochzählen den Inkrementoperator und zum Herunterzählen den Dekrementoperator.

Aufgabe 3: Halbwertszeit**3 Punkte**

Das radioaktive Nuklid Radium hat eine Halbwertszeit von 11,4 Tagen. Das bedeutet, dass sich nach dieser Zeit 50% der Atome in ein anderes Atom umgewandelt haben und nur 50% der Stoffmenge von dem Nuklid Radium übrig bleiben. Sobald das Nuklid Radium 98% der Stoffmenge verloren hat, gilt es als verbraucht. Schreiben Sie ein C-Programm, das berechnet wie viele Tage Radium gelagert werden kann, bis es 98% der Stoffmenge verloren hat. Zu Beginn liegen genau 200 mol von Radium vor. Nach 11,4 Tagen halbiert sich die Zahl der Teilchen auf 100 mol und nach 22,8 Tagen auf 50 mol u.s.w..

Für die Berechnung muss eine while-Schleife verwendet werden. Ihre formatierte¹ Ausgabe sollte etwa so lauten:

```
0.0 Tage: 200.0000 mol
11.4 Tage: 100.0000 mol
22.8 Tage:  50.0000 mol
34.2 Tage:  25.0000 mol
45.6 Tage:  12.5000 mol
57.0 Tage:   6.2500 mol
68.4 Tage:   3.1250 mol - VERBRAUCHT
```

Aufgabe 4: Check Epsilon**2 Punkte**

Fließkommatypen wie `float` und `double` repräsentieren Zahlen nur bis zu einer bestimmten Anzahl von Stellen hinterm Komma, welche *Genauigkeit* genannt wird. Addiert man zu einer solchen Variable z.B. eine sehr kleine Zahl `eps` (Epsilon) hinzu, ändert sich der ursprüngliche Wert nicht mehr. Der Typ der Variablen kann die Summe nicht mehr repräsentieren.

Schreiben Sie ein C-Programm, das die größtmögliche Zahl `eps` für den Datentyp `float` findet, welche unterhalb der Genauigkeitsschwelle für die Zahl 1000000 (1 Million) ist.

Oder mit anderen Worten: Finden Sie die größte Zahl, die klein genug ist, dass die Addition dieser mit 1000000 wieder 1000000 ergibt. Alle Zahlen sollen vom Typ `float` sein.

Der zu benutzende Algorithmus ist:

1. Setze `eps` auf 1 (vom Typ `float`)
2. Solange `1000000 (vom Typ float) + eps` ungleich `1000000 (vom Typ float)` ist: Teile `eps` durch 2
3. Gebe `eps` aus, nachdem die Schleife beendet ist.

¹C-Referenz zur print-format-Funktion: <http://www.cplusplus.com/reference/cstdio/printf/>