

Universität Potsdam  
Institut für Informatik

Sommersemester 2016

Praxis der Programmierung

8. Aufgabenblatt

1. Schreiben Sie ein C++-Programm, das den Benutzer auffordert, sein Alter einzugeben, dieses von der Standardeingabe entgegennimmt und wieder auf die Standardausgabe ausgibt:

```
Geben Sie Ihr Alter ein: | // Eingabe z.B. von "22"  
Sie sind 22 Jahre alt.
```

Realisieren Sie auch den konstanten Text und die Zeilenumbrüche. Benutzen Sie keine C-Bibliotheken.

*Hinweis: Benutzen Sie << zur Umlenkung des Datenstroms von der Tastatur (über cin) in eine Programmvariable. Die Umwandlung der Character-Folge in einen Zahltyp wird von den Funktionen aus iostream automatisch vorgenommen.*

2. Kopieren Sie die Datei `Point.cpp` und ergänzen Sie sie schrittweise wie folgt:
  - a) Definieren Sie eine Klasse `Point` von Punkten in der Ebene, die durch eine  $x$ - und eine  $y$ -Koordinate bestimmt sind. Die zugehörigen Datenelemente sind zu kapseln. Der Standardkonstruktor (parameterlose Konstruktor) erzeugt einen Punkt im Koordinatenursprung  $(0, 0)$ . Für jedes der Datenelemente (Koordinaten) gibt es eine `get`-Methode, die den aktuellen Wert des jeweiligen Datenelementes zurückgibt. Ferner gibt es eine Methode `moveTo(int, int)`, die einen Punkt auf als Parameter übergebene neue Koordinaten verschiebt.  
Wenn Sie alles richtig implementiert haben, ist die Datei kompilierbar und (mit der vorgegebenen `main`-Methode) ausführbar. Orientieren Sie sich beim Implementieren an der gegebenen `main`-Methode.
  - b) Ergänzen Sie jetzt einen weiteren Konstruktor mit zwei `int`-Parametern, der einen Punkt auf den als Parameter übergebenen Koordinaten erzeugt. Ergänzen Sie außerdem
    - eine Methode `moveRel(int, int)` zum Verschieben um die als Parameter übergebenen Werte und
    - eine Methode `distance()`, die den Abstand eines Punktes zum Koordinatenursprung berechnet und zurückgibt.

Ergänzen Sie die `main`-Methode sinnvoll, so dass alle neuen Elemente der Klasse getestet werden.

- c) Ergänzen Sie eine globale Funktion `reset(Point * p)`, die einen Pointer auf ein Exemplar der Klasse `Point` übergeben bekommt und dieses auf den Koordinatenursprung zurücksetzt.

Beachten Sie, dass eine globale Funktion nicht Methode der Klasse `Point` ist. Trotzdem muss der Zugriff auf die privaten Datenelemente ermöglicht werden. Wie gehen Sie vor? Hierfür können Sie sich an folgendem Beispiel aus der Vorlesung orientieren:

```
class Date {
    public: Date();
    private: int day, month, year;
    friend void setNewYear(Date * d, int year);
};

Date::Date() { day = 1; month = 1; year = 1970; }

void setNewYear(Date * d, int year) { d->year = year; }

int main() {
    Date nextNewYear; // Aufruf des Standardkonstruktors
    setNewYear(&nextNewYear, 2016);
}
```

*Begründen Sie:* Weshalb kann `setNewYear` statt eines Pointers nicht das `Date`-Objekt selbst übergeben werden?