

Fakultative Aufgaben GK Informatik
spätester Abgabetermin ist Freitag, der 5. November 2016

Die Aufgaben werden wie eine Leistungskontrolle gewertet und gehen je nach deiner Wahl in die Wertung 12/1 oder 12/2 ein, alternativ kann eine Leistungskontrolle im Nachhinein gestrichen werden.

Erstellen Sie ihr Programm mit BlueJ, nutzen Sie die Klasse InOut.
Erläutern Sie die Funktion der Programmteile durch Kommentare direkt im Quelltext.
Dokumentieren Sie in einer Textdatei in Kurzform die Bedienung des Programms.

Jede der Aufgaben 1.-5. kann in der Regel nur einmal abgegeben werden, also sprechen Sie sich rechtzeitig mit mir ab, ... wer zuerst kommt...

Kopien von Lösungen anderer oder aus dem Netz sind sinnlos, ich merke das und werte Sie dann nicht.

Im Zweifelsfall löchere ich Sie mit Fragen zum Programm, und Sie können mich von der Eigenleistung überzeugen...

Auch unperfekte Lösungen können abgegeben werden, dann gibt es eben statt einer Note Zusatzpunkte für eine LK oder Klausur.

Sie haben die Wahl zwischen folgenden Aufgaben:

1. Caesar und Vigenère – Verschlüsselung,
2. Entweder Caesar oder Vigenère, dann aber sowohl Ver- als auch Entschlüsselung
3. Erkennen, ob ein eingegebenes Wort bzw. ein Satz ein Palindrom ist.
(*Ein Neger mit Gazelle zagt im Regen nie.*)
4. In einem beliebigen Text die angegebenen Werte von km in Meilen umrechnen bzw. bei Angaben unter 0,5km in Yard. (siehe Navigationsgerätaufgabe, nur dass der Zahlenwert jetzt erst aus einem Text extrahiert werden soll und dann wieder in den Text eingefügt wird.
5. Eine Dezimalzahl in Dualzahlen umrechnen und umgekehrt.
6. Eine Dezimalzahl in Hexadezimalzahlen umrechnen und umgekehrt.
7. Schatzsuche: Diese Aufgabe kann auch als Partnerarbeit bearbeitet werden, wenn dazu strikt objektorientiert gearbeitet wird. Interessant wäre es auch, die Bibliothek Stifte & Mäuse zu benutzen.
Ziel des Spieles ist es, den zu Spielbeginn (zufällig?) auf dem Spielfeld platzierten Schatz zu finden. Es gibt mindestens diese vier Klassen: Die Klasse **Spiel** [zum Spielen], welche die Klassen **Troll**, **Schatz** und **Spielfeld** aufruft.

0,0	1,0					X,0
0,1	hoch ↑					
zurück ←		→	vor			
	↓					
	runter					
						
0,Y						X,Y

- Die Größe des **Spielfeldes** wird zu Beginn festgelegt.
- Der **Troll** fragt dich nach seinem Namen und an welcher Stelle er starten soll.
- Trolle sind doof und laufen im Spielfeld ziellos umher. Er kann nur jeweils: einen Schritt vor, zurück, hoch, runter. Nach jedem Schritt gibt der Troll zur Kontrolle seine Position an.
- Der **Schatz** will nur seine Position speichern.

Es muss auch kontrolliert werden, ob die Eingaben sinnvoll sind, also ob Troll und Schatz innerhalb des Spielfeldes liegen. Wenn nicht, wird das Spiel beendet. *Befehl: System.exit (0);*

Dann wird gespielt. D.h. der Troll läuft je nach Eingabe in eine der vier Richtungen. Wenn der Troll an einen Rand des Spielfeldes gelangt, kommt die Ausgabe „Geht nicht weiter, du bist am (rechten/linken/oberen/unteren) Rand“.

Hat er das Feld mit dem Schatz gefunden, dann sagt er „*Juhu, [Troll-Name] hat den Schatz gefunden!*“ und das Spiel ist beendet.