

Name: _____ Tutor: _____

Matrikelnummer: _____ Punkte: _____

Gruppe: _____ Abzugeben bis: 22.11.00, 12:00

Übungsleiter: _____ Bearbeitungsdauer: _____

Aufgabe 1: Attack at dawn! _____ 10 Punkte

Schon Cäsar wußte um die Bedeutung der Kryptographie, und so geht die Geschichte, daß er einen Angriffsbefehl verschlüsselt an seine Befehlshaber sandte. Er verwendete dabei ein einfaches Verfahren, die sogenannte Cäsar-Chiffre. Dabei wird das Alphabet so verschoben, daß ein „c“ die Bedeutung eines „a“ hat, ein „d“ die eines „b“ usw. Siehe dazu auch folgende Darstellung:

Klartext: a b c d e f g h i j k l m n o p q r s t u v w x y z
 verschlüsselt: c d e f g h i j k l m n o p q r s t u v w x y z a b

Schreiben Sie ein Programm, das einen Buchstaben einliest, der die Stelle des „a“ erhalten soll und danach einen Text, der verschlüsselt werden soll.

Bei der (beispielhaften) Eingabe „c Attack at dawn!#“ sollte das Programm 'c' als den Schlüssel betrachten, das Leerzeichen überlesen und dann den Text verschlüsselt, also „Cvvcem cv fcyp!“, ausgeben. Um das Programm leichter prüfen zu können, geben Sie Ein- und Ausgabebetext am besten so aus wie in der Darstellung rechts gezeigt.

Schlüssel: c	
A	C
t	v
t	v
a	c
c	e
k	m
a	c
t	v
d	f
a	c
w	y
n	p
!	!
#	

Hinweise

- Verwenden Sie zeichenweise Eingabe.
- Satzzeichen sowie Leerzeichen werden nicht verschlüsselt.
- Ein „#“ am Ende des Eingabetextes signalisiert das Ende des zu verschlüsselnden Textes. Dieses Zeichen wird nicht mehr verschlüsselt.
- Testen Sie das Programm auch mit oben angeführtem Text.
- Das Programm kann mehrere Texte verschlüsseln (und nicht nur 'c' als Schlüssel verwenden :)

Aufgabe 2: Annäherungen _____ 14 Punkte

Die Werte der Funktionen "Sinus" und "Kosinus" können für ein gegebenes x jeweils durch folgende Taylor-Reihen angenähert werden:

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots \text{ und } \cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots \text{ mit: } \left(-\frac{\pi}{2} \leq x \leq \frac{\pi}{2}\right)$$

Erstellen Sie ein Java-Programm das Tabellen für Kosinus-Werte unter Anwendung der Taylor-Reihe berechnet und ausdrückt. Der Startwert (in rad), Endwert (in rad) und die Schrittweite der Tabelleneinträge (in rad) werden eingelesen. Start- und Endwerte müssen im Bereich [0 - 2π] liegen.

Bonus

Wenn Sie die Aufgabe um die Berechnung des Sinus erweitern, erhalten Sie 2 Bonuspunkte.

Hinweise

- Winkel in rad erstrecken sich von 0 bis 2π.
- Sollten Sie die Absolutbetrag eines Wertes brauchen (z.B. |-5|), können Sie dazu folgenden Aufruf verwenden: Math.abs() mit z.B. Math.abs(-5) liefert 5.
- Die Verwendung der Methoden sin und cos aus der Math-Klasse ist nicht zulässig - höchstens für Kontrollzwecke.
- Beachten Sie, dass Sie bei einer Näherung mit einer Fehlerschranke arbeiten müssen.
- Achten Sie auf die Wahl des richtigen Datentyps.
- Ein Bildschirmdialog könnte z.B. so aussehen (wobei mit einer Fehlerschranke von 0.001 gearbeitet wurde und in der ganz rechten Spalte die Ergebnisse von Math.cos() zum Vergleich ausgegeben wurden):

Bitte Startwert (in rad) eingeben: 0.0
Bitte Endwert (in rad) eingeben: -1.0
Bitte Schrittweite eingeben: 0.1
Falsche Parametereingabe.

Nochmal? (j/n)

Bitte Startwert (in rad) eingeben: 0.0
Bitte Endwert (in rad) eingeben: 6.23
Bitte Schrittweite eingeben: 1.0

1	0.0	1.0		1.0
2	1.0	0.5403025793650793		0.5403023058681398
3	2.0	-0.4161552028218696		-0.4161468365471424
4	3.0	-0.989994494902419		-0.9899924966004454
5	4.0	-0.6536333241518434		-0.6536436208636119
6	5.0	0.283625015089173		0.28366218546322625
7	6.0	0.9601630532594905		0.960170286650366

Nochmal? (j/n)