Prozedurale Programmierung

Aufgaben 3
Kontrollstrukturen

1. Maximumsberechnung

a) Nach der Eingabe zweier reellen Zahlen ist mit Hilfe einer if-Anweisung das Maximum zu bestimmen und auszugeben.

b) Bestimmen Sie nach einer Programmerweiterung das Maximum von drei beliebigen reellen Zahlen, wobei zur Problemlösung eine geschachtelte if-Anweisung eingesetzt werden soll.

Bei Gleichheit der drei Zahlen soll zusätzlich eine Ausschrift „Zahlen gleich“ ausgegeben werden.

2. Berechnen des Funktionswertes einer Funktion

Eine Funktion y=f(x) sei definiert durch

$$y = \begin{cases} \dfrac{1}{1+x^2} & \textit{für} \quad x \le 0 \\ 1-\dfrac{x^2}{10} & \textit{für} \quad 0 < x \le 5 \\ \dfrac{15}{4}x & \textit{für} \quad 5 < x \le 30 \\ \dfrac{1}{8} & \textit{für} \quad x > 30 \end{cases}$$

Ein beliebiger Argumentwert x soll eingelesen, sein Funktionswert y berechnet und die Werte x und y ausgegeben werden.

3. Schreiben Sie ein Programm für die Nachbildung eines "Minicomputers", der einfache arithmetische Ausdrücke der Form

<operand_1> <operator> <operand_2>

berechnet.

Einzugeben sind: operand_1, operator und operand_2

Zugelassene Operatoren:	Addition	+
	Subtraktion	-
	Multiplikation	*
	Division	/ oder :

Bei Division durch Null bzw. unzulässigem Operator ist eine Fehlerausschrift auszugeben. Die Ergebnisausgabe soll im Programm nur einmal erfolgen.

4. Bestimmen des Maximums n beliebiger Zahlen

- Anzahl der Zahlen n und den ersten x-Wert der Zahlenfolge einlesen
- Der Variable Maximum x-Wert zuweisen
- Weitere x-Werte einlesen, die x-Werte mit dem Maximum vergleichen, Maximum gegebenenfalls mit dem größeren x überschreiben

Testen Sie mit folgenden Zahlenpaaren (2, 5, 3, 3, 2), (-2, -4, -1)

5 . Berechnung der Exponentialfunktion exp(x) näherungsweise mit Hilfe der TAYLOR-Reihe

$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \ldots + \frac{x^n}{n!} + \ldots$ (konvergiert für $|x|<\infty$).

- Der x-Wert ist einzulesen
- Die Berechnung ist abzubrechen, wenn das nächste Glied der Reihe betragsmäßig kleiner als eine einzulesende Genauigkeitsschranke (e<1) ist
- Bei Aufstellung des Lösungsalgorithmus ist zu beachten, dass es nicht optimal ist, die einzelnen Reihenglieder völlig unabhängig voneinander zu berechnen.

  Mit g0=1 (erstes Glied) gilt $g_i = g_{i-1} \cdot \frac{x}{i}$, für i=1,2,…

6. Es soll folgendes Spiel durch ein Computerprogramm simuliert werden: Der Spieler zahlt für jedes Spiel einen Einsatz von 1 Euro. Er darf einmal mit 3 Würfeln würfeln. Je nach gewürfelter Punktzahl bekommt der Spieler einen Gewinn entsprechend der Tabelle ausgezahlt.

Punktzahl	Gewinn in €
3 – 14	0,-
15	2,-
16	5,-
17	10,-
18	100,-

Simulieren Sie 1000 Spiele mit zufälligen Würfen und geben Sie das Ergebnis der Gewinn-/Verlustrechnung aus.
Hinweis: Die Funktion rand() (Header <cstdlib> einbinden) liefert Zufallszahlen zwischen 0 und **RAND_MAX**. Zufallszahlen von 0 bis 5 bekommen Sie, indem Sie den Rest der Division durch 6 bestimmen: rand()%6.
Mit dem Aufruf srand((unsigned)time(NULL)); kann der Zufallszahlengenerator initialisiert werden (Header <ctime> einbinden).