

### Aufgabe 1.2

1. Nimm Zirkel und stich in Punkt B ein. Zeichne nun einen Kreis K1 um B mit Radius  $r_1$ , der dem Abstand der beiden Punkte A und B ( $r_1 = d(A,B)$ ) entspricht.
2. Nimm Zirkel und stich in Punkt A ein. Zeichne nun einen Kreis K2 um A mit Radius  $r_1$ , der dem Abstand der beiden Punkte A und B ( $r_1 = d(A,B)$ ) entspricht.
3. Zeichne eine Gerade g durch die beiden Schnittpunkte der Kreise K1 und K2 mit Hilfe eines Lineals.
4. Nimm Zirkel und stich in Punkt B ein. Zeichne nun einen Kreis K3 um B mit Radius  $r_2$ , der dem Abstand der beiden Punkte C und B ( $r_2 = d(C,B)$ ) entspricht.
5. Nimm Zirkel und stich in Punkt C ein. Zeichne nun einen Kreis K4 um C mit Radius  $r_2$ , der dem Abstand der beiden Punkte C und B ( $r_2 = d(C,B)$ ) entspricht.
6. Zeichne eine Gerade h durch die beiden Schnittpunkte der Kreise K3 und K4 mit Hilfe eines Lineals.
7. M sei der Schnittpunkt der Geraden g und h. Stich nun mit Zirkel in M ein und zeichne einen Kreis k mit dem Radius  $r_3$ , der dem Abstand der Punkte M und A entspricht ( $r_3 = d(M,A)$ ).

=> k ist der Umkreis des Dreiecks ABC.

### Aufgabe 1.3

a)  $(16)_{10} = 1 \cdot 16^1 + 0 \cdot 16^0 = (10)_{16}$

b)  $(fb)_{16} = (1111\ 1011)_2$

$(f)_{16}$  entspricht  $(1111)_2$ ;  $(b)_{16}$  entspricht  $(1011)_2$

„Erklärung“: Binärzahl in Blöcke zu je 4 „Bit“ aufgeteilt ( $2^4 = 16$ )

$$c) (13967)_{10} = 3 \cdot 16^3 + 6 \cdot 16^2 + 8 \cdot 16^1 + 15 \cdot 16^0 = (36 \ 8f)_{16}$$
$$12288 + 1536 + 128 + 15$$

größtes  $n$  für  $16^n < 13967$ :  $16^3 (= 4096)$

Hinweis: „15“ entspricht im Hexadezimalsystem/Sedezimalsystem „f“

$$d) (0.25)_{10} = 0 \cdot 2^{-1} + 1 \cdot 2^{-2} = (0.01)_2$$

$$e) \begin{matrix} 1001 & 0011 & 0111 & 0101 \end{matrix}_2 = (9375)_{16}$$
$$\begin{matrix} 9 & 3 & 7 & 5 \end{matrix}$$

#### Aufgabe 1.4

a) Determinismus: Hat ein Algorithmus bei mehrfacher Ausführung für den gleichen Eingabeparameter immer das gleiche Ergebnis liefert, so hat dieser Algorithmus ein deterministisches Ergebnis. Ist der Ablauf des Algorithmus jedes Mal gleich, hat der Algorithmus einen deterministischen Ablauf.

b) Sequenz (Reihe/„Hintereinanderausführung“): Im Normalfall sind die verschiedenen Befehle (oft durch Semikolon getrennt) in der gegebenen Reihenfolge auszuführen (von links nach rechts und von oben nach unten).

c) Beim Imperativen Programmieren haben die Verfahrensanweisungen einen befehlenden (imperativen) Charakter, der Programmierer schreibt genau vor, welche Arbeitsschritte in welcher Reihenfolge durchgeführt werden sollen. Eine imperative Programmiersprache ist sozusagen das Bindeglied zwischen der menschlichen Denkweise (Problemstrukturen und Begrifflichkeit der *Problemdomäne*) und den Maschinenbefehlen des von-Neumann-Rechners, die nur auf die Rechnerarchitektur ausgerichtet sind (und nichts mit dem zu lösenden Problem an sich zu tun haben).

d) Der Entscheidungsgehalt  $H$  eines Zeichenvorrats mit  $n$  Zeichen ist die kleinste Anzahl  $H$  von Entscheidungen, mit denen man feststellen kann, welches der  $n$  Zeichen vorliegt. Der Entscheidungsgehalt wird in bits ( $b$ ) gemessen.

e) Im Hexadezimalsystem werden Zahlen in einem Stellenwertsystem zur Basis 16 dargestellt. Zeichenvorrat:  $\{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F \}$

f) Variablen sind Speicher für Zwischenergebnisse, die sich während des Programmablaufs verändern können, d.h. ihnen kann zum Beispiel ein neuer Wert zugewiesen werden. In Variablen können je nach Bedarf zum Beispiel natürliche Zahlen, Fließkommazahlen oder Zeichenketten gespeichert werden.

Ein Algorithmus mit deterministischem Ergebnis muss nicht immer einen deterministischen Ablauf haben.

Ein Beispiel hierfür ist die Konstruktion des Mittelpunktes  $M$  eines gegebenen Kreises  $k$ . Man wählt 3 **beliebige** (nicht identische) Punkte  $A$ ,  $B$  und  $C$  auf einem Kreis und verbindet  $A$  mit  $B$  und  $B$  mit  $C$ . Anschließend konstruiert man je eine Senkrechte auf dem Mittelpunkt der Strecke  $AB$  und eine auf dem Mittelpunkt der Strecke  $BC$ . Der Schnittpunkt der beiden Senkrechten ist  $M$ . Da dies mit beliebigen Punkten auf  $k$  funktioniert ist der Ablauf nicht deterministisch.

Unterschied zw. Validierung, Verifikation und Test:

Die Validierung ist die Überprüfung, ob die Spezifikation wirklich das Problem und nicht mehr und nicht weniger *beschreibt*.

Die Verifikation ist der Nachweis, dass der Algorithmus das spezifizierte Problem korrekt löst.

Mit dem Test überprüft man dann letztendlich, ob das Programm den Algorithmus auch korrekt umsetzt.