

4. Wenn Sie hier ankommen, haben Sie alle noch freien Karten ausprobiert und in den Papierkorb geworfen. Also sind Sie mit der zuletzt gelegten Karte (nennen wir sie X) in eine Sackgasse geraten. Sie gehen in der Rekursion eine Stufe zurück, d.h. Sie stellen die Situation wieder her, wie sie war, als Sie X legten, und befinden sich nun mitten in Punkt 2. Diese Karte X, von der Sie glaubten, sie passe, war nämlich doch nicht die richtige. (Zu 2.: Falls Sie X noch nicht dreimal gedreht haben, versuchen Sie es weiter mit X, sonst legen Sie X in den Papierkorb und nehmen die nächste Karte.)
- +. Wenn Sie nicht nur eine, sondern alle Lösungen finden wollen, hören Sie in 2. einfach nicht auf, sondern fahren bei 4. weiter.

An meisten Mühe, abgesehen von der schier unendlichen Dauer der Suche, macht uns Menschen wohl die Rekursion in den Punkten 2. und 4., die sich nur schwer in unser sequentielles Denkschema pressen lässt. Wer sie nicht kennt, dem ist sie zuerst etwas unheimlich - man versteht nicht recht, wieso das funktionieren soll. Doch mit der Zeit gewinnt man Vertrauen und lernt die Eleganz der rekursiven Problemlösung schätzen.

Implementation in LOGO

Die Programmiersprache LOGO eignet sich hervorragend zur Implementation von rekursiven Algorithmen. Da es nur dynamische Datenstrukturen gibt, sind diese sehr gut unterstützt (hier Kartenstapel, die wachsen und abnehmen können und manchmal auch zusammengefügt werden). (Die Nummern in eckigen Klammern verweisen im folgenden auf die entsprechenden Stellen in der Programmliste.)

Eine wichtige Vorentscheidung fällt bei der Wahl der Datenstruktur: Die Liste :schildkroeten [1] dient als Eingabe für das Programm und enthält neun Unterlisten, für jede Karte eine. Die Reihenfolge entspricht genau der Zeichnung im Bulletin Nr. 3. Jede Teilliste enthält die Nummer der betreffenden Karte, die Anzahl 900-Drehungen, die damit ausgeführt wurden (hier überall 0) und eine weitere Unterliste, die die Schildkrötenanteile auf dieser Karte beschreibt. Beispiel: Auf der zweiten Karte befindet sich oben der 1. (vordere) Teil einer ausgefüllten Schildkröte, sowie im Uhrzeigersinn der 1. Teil einer karierten, der 2. (hintere) Teil einer gepupften und der 2. Teil einer gestrichelten.

Das Hauptprogramm suche [2] wird mit suche :schildkroeten aufgerufen und ruft im wesentlichen die Prozedur suchen [3,4] auf, mit der Eingabeliste und zwei leeren Listen.

suchen [4] hat drei Parameter, die alle Listen der Art von :schildkroeten sind; :vorrat enthält die Karten, die noch nicht