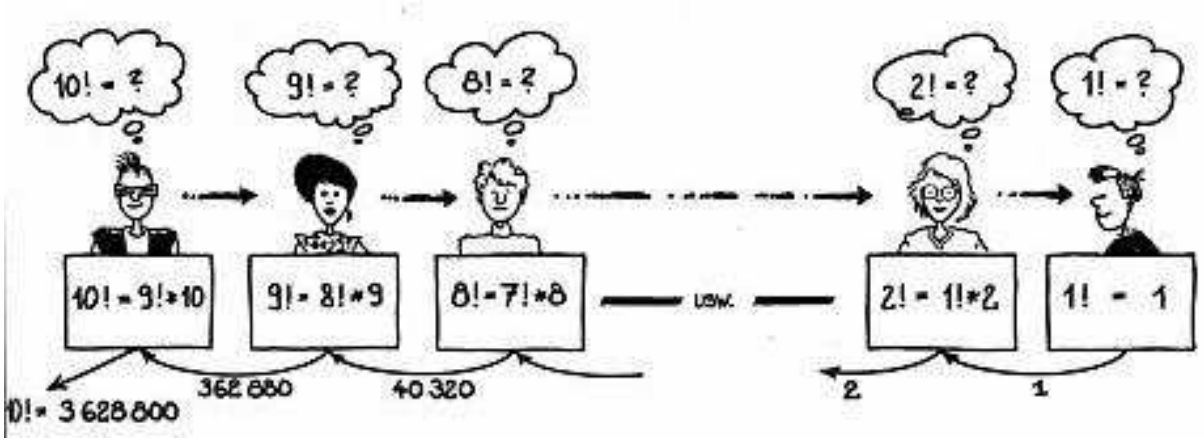
Rekursion

Die **Rekursion** ist eine wichtige Strategie zum Lösen komplizierter Probleme. Dabei wird das zu lösende Problem auf ein gleiches Problem mit veränderten Argumenten zurückgeführt bis das zurückgeführte Problem einfach gelöst werden kann (*Abbruchbedingung*). Bei einer **echt-rekursiven Funktion** (oder Prozedur) wird anschließend aus dieser Lösung die Lösung des Ausgangsproblems ermittelt.

Beispiel: Berechnung der Fakultät einer natürlichen Zahl n

Für eine natürliche Zahl n > 0 ist n! = n * (n – 1) * (n – 2) * ... * 2 * 1
Die Berechnung von n! ergibt also das Produkt aller natürlicher Zahlen von 1 bis einschließlich n.
Dabei gelten folgende Berechnungsvorschriften: n! = n * (n – 1)! und 1! = 1.
Eine Lösungsstrategie für das Bestimmen von n! geht aus der folgenden Abbildung hervor[1]:



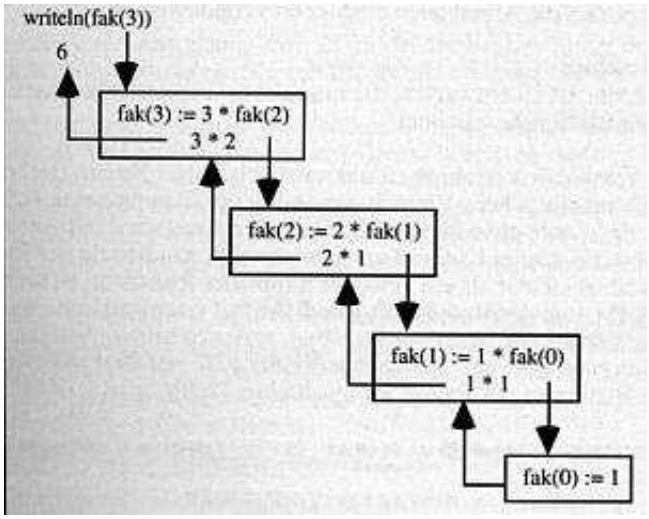
Zu berechnen ist 10!. Zur Lösung der Aufgabe werden mehrere Personen befragt. Die erste Person kann 10! leider nicht berechnen, aber sie kennt die Bildungsvorschrift. Laut dieser muss sich 10! als 10 * 9! berechnen lassen. Die Berechnung von 9! wird an die nächste Person weitergereicht. Sie kann 9! leider auch nicht berechnen, sondern kennt wiederum nur die Bildungsvorschrift 9! = 9 * 8! usw. Die letzte befragte Person hat dann 1! zu berechnen. Das Ergebnis für dies Rechnung ist aber laut Definition bekannt: Es ergibt sich 1. Dieses Ergebnis wird nun an die vorherige Person zurückgereicht. Nun kann sie 2! berechnen, da ihr das Ergebnis von 1! bekannt ist. Diese Kette setzt sich bis an den Anfang fort und somit ergibt sich 10! = 10 * 9 * 8 * ... * 2 * 1 = 3628000.

Soweit zur anschaulichen Erklärung und nun zur Theorie.

Wenn n = 1 ist, ergibt sich also laut Definition der Fakultät der Wert 1 → **Abbruchbedingung**. Wenn n ≥ 1 gilt, so ist zu rechnen n * (n – 1)! → **Rekursionsschritt**.
Zunächst wird das Problem mit Hilfe des Rekursionsschritts auf ein einfacheres Problem zurückgeführt, bis man auf ein einfach lösbares Problem stößt (**rekursiver Abstieg**). Die Berechnung des Endergebnisses erfolgt dann beim **rekursiven Aufstieg** aus der **Rekursionstie-**

[1] Junger u.a.: *Informatik*, Diesterweg/Sauerländer, 1988

fe. Die nächste Abbildung soll diesen Vorgang noch einmal am Beispiel der Berechnung von 3! verdeutlichen.[2]



Für die Entwicklung eines rekursiven Lösungsverfahrens sind also folgende Fragen wichtig und zu beantworten:

1. Wie lautet eine einfachere Version des gegebenen Problems? (**rekursiver Abstieg**)
2. Wodurch ist der direkt lösbare Fall gekennzeichnet? (**Abbruchbedingung**)
3. Wie gewinnt man aus der Lösung des einfachen Problems die Lösung des ursprünglichen Problems? (**rekursiver Aufstieg**)

Die rekursive Funktion zur Berechnung der Fakultät lautet somit folgendermaßen:

```
function fak_rekursiv(n: integer): integer;
begin
  if n = 1 then fak_rekursiv := 1   // Abbruchbedingung
    else fak_rekursiv := n * fak_rekursiv(n - 1); // Rekursionsschritt
end;
```

[2] Baumann, Rüdeger: *Informatik für die Sekundarstufe II*, Band 1, Ernst Klett Schulbuchverlag, Stuttgart, 1992