

### 1. Rekursive Berechnung der Summe (3 Punkte)

Entwickeln Sie eine rekursive Funktion, die die Summe der ersten  $n$  Zahlen nach der angegebenen Formel bestimmt

$$\text{sum}(n) = \begin{cases} 0 & \text{falls } n = 0 \quad (\text{Rekursionsanfang}) \\ \text{sum}(n-1) + n & \text{falls } n \geq 1 \quad (\text{Rekursionsschritt}) \end{cases}$$

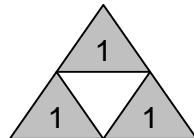
### 2. Pascal'sches Dreieck (10 Punkte)

Das Pascal'sche Dreieck ist ein Beispiel für eine Rekursion. Ausgehend von einem einzelnen gleichseitigen Dreieck wird durch sukzessive Erweiterung d.h. hinzufügen neuer Zeilen, die jeweils ein Dreieck mehr als die vorherige enthält, ein neues gleichseitiges Dreieck erzeugt.

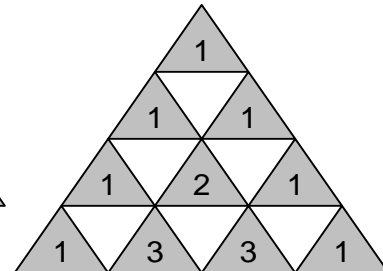
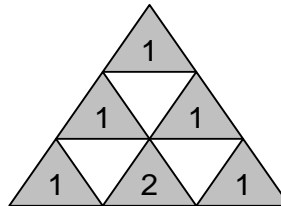
Anfangssituation 1. Erweiterung



2. Erweiterung



3. Erweiterung



Jedem Dreieck ist ein numerischer Wert zugeordnet. Dreiecke am Rand haben den Wert 1, andere berechnen ihren Wert aus der Summe der Nachbarn in der vorherigen Zeile.

Mathematisch lässt sich der Zusammenhang wie folgt darstellen:

$$C(n, k) = C(n-1, k-1) + C(n-1, k), n \dots \text{aktuelle Zeile}, k \dots \text{aktuelle Spalte}$$

Entwickeln Sie eine rekursive Lösung für dieses Problem. Als Abbruchbedingung soll die Anzahl der Erweiterungen dienen. Geben Sie die berechneten Werte in „ansprechender“ Form am Bildschirm aus. Beachten Sie, dass die Werte des Dreiecks nicht gespeichert werden dürfen.

### 3. Strukturen in C (7 Punkte)

Neben skalaren Datentypen gibt es auch strukturierte Datentypen wie Felder und Strukturen (`struct`). Felder sind Ihnen hinlänglich bekannt. Strukturen wurden bis jetzt nicht behandelt. Erarbeiten Sie sich das Thema selbständig und bereiten Sie dieses so vor, dass Sie Ihren Mitstudierenden die wesentlichen Aspekte dazu näherbringen können. Zeigen Sie die verschiedenen Aspekte anhand von kleinen Codebeispielen.