Aufgabe 1:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Operation | Einfache Arrays | Arrays mit zyklischer Speicherung | Einfach verkettete Listen | Mehrfach verkettete Listen |
| Einfügen am Anfang | O(n) | O(n) | O(1) | O(1) |
| Einfügen am Ende | O(n) | O(1) | O(1) | O(1) |
| Einfügen an einer Position p | O(n) | O(n) | O(1) | O(1) |
| Suchen eines Elements mit gegebenem Wert | O(n) | O(n) | O(n) | O(n) |
| Hintereinanderhängen zweier Listen | O(n) | O(1) | O(1) | O(1) |
| Umkehren einer Liste | O(n) | O(1) | O(n) | O(1) |

Aufgabe 2:

1. Es gilt folgende Formel: a(n) = n\*a(n-1) + 1, a(0) = 0 // Das Ergebnis sind die Aufrufe von perm() insgesamt

"Summe der Länge der ersten Durchläufe in allen Permutationen von [n].

Beispiel: a(3) = 10 weil die Längen von den ersten Durchläufen in den Permutationen (123), (13)2, (3)12, (2)13, (23)1 und (3)21: 3, 2, 1, 1 2 und 1 sind.

Jeweils sind die ersten Durchläufe eingeklammert. Die Anzahl der Zellen in den letzten Spalten von allen Deco-Polyominoes der Höhe n. Ein Deco-Polyomino ist ein gerichteter spalten-konvex Polyomino in dem die Höhe, gemessen entlang der Diagonalen, nur in der letzten Spalte erreicht wird. a(n) = Sum(k\*[A092582](https://oeis.org/A092582)(n,k), k = 1..n)." (Englischer Originaltext von Emeric Deutsch, Quelle: <https://oeis.org/search?q=0%2C+1%2C+3%2C+10%2C+41%2C+206%2C+1237&language=german&go=Suche>)

2. Im Durchschnitt entfallen a(n – 1) + (bei sehr großen Zahlen ist diese Addition quasi kaum merkbar, Bsp. n = 1.000.000 => + 1/1.000.000 daher optional/weglassbar) Aufrufe auf eine einzelne Permutation.