



1. Die C-Funktion (leicht) (4)

Schreiben Sie eine Funktion, die für eine gegebene Permutation $P \in \Sigma^*$ der Länge n den Wert der C-Funktion $C : \Sigma^* \rightarrow \mathbb{Q}$ ermittelt .

2. Nächste Permutation (mittel) (5)

Implementieren Sie eine Funktion **nextPerm** die bei einer gegebenen Permutation P_i die nachfolgende Permutation P_{i+1} berechnet, wobei gilt $P_{i+1} > P_i$ (wenn man die Permutation als Zahl mit den Ziffern $1, 2, \dots, n$ auffasst. Gehen Sie dabei wie folgt vor. Gehen Sie die Elemente der Permutation von hinten nach vorne durch, bis Sie zu einem Element kommen, das kleiner als das zuvor besuchte Element ist. Vertauschen Sie nun dieses Element mit dem nächst größeren Element im "rechten" Teil der Permutation. Anschliessend spiegeln Sie die "rechts" stehenden Elemente.

Beispiel:

5 1 4 3 7 6 2

5 1 4 (3) 7 6 2
←

5 1 4 (3) 7 (6) 2
↷

5 1 4 6 (7 3 2)
↔

5 1 4 6 (2 3 7)

3. Suche des maximalen Wertes der C-Funktion für n Elemente (mittel) (4)

Es sei $P_{max}^{(n)}$ eine Permutation von n Elementen, deren C-Wert maximal bezüglich aller anderen solchen Permutationen ist, d.h. $C(P_{max}^{(n)}) \geq C(P), \forall P \in S_n$. Schreiben Sie nun ein Programm, das mittels der in den beiden vorherigen anderen Aufgaben entwickelten Funktionen für eine vom Benutzer zu erfragende Länge n

eine Permutation $P_{max}^{(n)}$ von n Elementen ermittelt. Geben Sie Testläufe für $n = 3$ bis 9 ab.

Anmerkungen:

S_n = Menge der Permutationen der ersten n Zahlen. Eine Permutation $P \in S_n$, $P = (1 \rightarrow P_1, 2 \rightarrow P_2, \dots, n \rightarrow P_n)$ wird in der Regel als Wort $P_1P_2..P_n$ geschrieben, d.h. $P \in \Sigma^*$ mit $\Sigma = 1, 2, \dots, n$ und alle Zahlen treten in P genau einmal auf.

Wenn Sie Ihre Abgaben korrigiert haben möchten, übersenden Sie diese immer bis zum Dienstag vor der Übung um 10.00 Uhr an sascha.riexinger@fmi.uni-stuttgart.de.