

Aufgabe zum Programmwurf zur Vorlesung „Grundlegende Algorithmen und Verfahren der KI“

Aufgabe 12

Das Partitionsproblem ist algorithmisch schwer zu lösen ist (exponentielle Laufzeit). Gegeben ist eine Menge von N Objekten mit den Größen $a_1, a_2, \dots, a_N \in \mathbb{N}$. Gesucht ist eine Teilmenge $P \subseteq \{1, 2, \dots, N\}$ der Objekte, für welche $\sum_{i \in P} a_i = \sum_{i \notin P} a_i$ gilt.

Mit einem evolutionären Algorithmus soll eine derartige Partition P gefunden werden. Für die Erstellung einer Implementierung sind folgende Anforderungen zu beachten.

- Ausgangspunkt ist eine Folge von Größen $a_1, a_2, \dots, a_N \in \mathbb{N}$.
 - Als genetische Repräsentation eines Lösungsvorschlags dient eine binäre Folge der Länge N . Dabei gilt $i \in P$ genau dann, wenn die i -te Stelle der Folge eine 1 ist.
 - Implementieren Sie die Transformationen Selektion, Mutation und Rekombination.
 - Wählen Sie $N \geq 10$ und starten Sie mit einer initialen Population von mindestens 50 Individuen.
- (a) Erstellen Sie eine Implementierung des Problems mit einem evolutionären Algorithmus unter Beachtung der Anforderungen.
- (b) Geben Sie eine Beschreibung (Dokumentation) der implementierten Funktionalität an.
- (c) Überlegen Sie sich ein Testszenario und führen Sie die Tests durch. Bewerten Sie die Ergebnisse.
- (d) Lassen Sie die Evolution über eine variable Anzahl an Generationen laufen, um eine Lösung des Problems zu finden. Dokumentieren und interpretieren Sie die erhaltenen Ergebnisse.