Aufgabe 2

Ein Raumausstattungsunternehmen steht immer wieder vor dem Problem, feststellen zu müssen, ob ein gegebener rechteckiger Fußboden mit rechteckigen Teppichresten ohne Verschnitt ausgelegt werden kann. Alle Längen sind hier ganzzahlige Meterbeträge. Haben sie beispielsweise zwei Reste der Größen 3×5 und einen Rest der Größe 2×5, so kann ein Fußboden der Größe 8×5 ausgelegt werden.

Das Unternehmen beauftragt eine Softwarefirma mit der Entwicklung eines Programms, welches diese Frage für beliebige Größen von Fußboden und Teppichresten entscheiden soll. Bei der Abnahme weist die Softwarefirma darauf hin, dass das Programm im wesentlichen alle Möglichkeiten durchprobiert und daher für große Eingaben schnell ineffizient wird. Auf die Frage, ob man das nicht besser machen könne, lautet die Antwort, dass das vorgelegte Problem NP-vollständig sei und daher nach derzeitigem Kenntnisstand der theoretischen Informatik nicht mehr zu erwarten sei.

> **Exkurs:** `SUBSET SUM`
> Das Teilsummenproblem (`SUBSET SUM` oder `SSP`) ist ein spezielles Rucksackproblem.
>
> Gegeben sei eine Menge von ganzen Zahlen $I = \{w_1, w_2, \ldots, w_n\}$. Gesucht ist eine Untermenge, deren Elementsumme maximal, aber nicht größer als eine gegebene obere Schranke c ist.

(a) Fixieren Sie ein geeignetes Format für Instanzen des Problems und geben Sie konkret an, wie die obige Beispielinstanz in diesem Format aussieht.

> Problem L
>
> **1. Alternative** $I = \{x_1, y_1, \ldots, x_n, y_n, c_x, c_y\}$
>
> **2. Alternative** $I = \{w_1, \ldots, w_n, c\}$

(b) Begründen Sie, dass das Problem in NP liegt.

> Es existiert ein nichtdeterministischer Algorithmus der das Problem in Polynomialzeit entscheidet:
>
> - nichtdeterministisch Untermenge raten ($\mathcal{O}(n)$)
> - Prüfe: ($\mathcal{O}(n)$)
>
> **1. Alternative** Elementsumme der Produkte (x_i, y_i) aus Untermenge $= c$
>
> **2. Alternative** Elementsumme der Untermenge $= c$

(c) Begründen Sie, dass das Problem NP-schwer ist durch Reduktion vom NP-vollständigen Problem SUBSET-SUM.

$\texttt{SUBSET SUM} \preceq_p \texttt{L}$

1. Alternative Die Funktion f ersetzt jedes w i durch w i , 1 und c durch c, 1 und startet TM für L

Berechenbarkeit: Hinzufügen von 1 für jedes Element, offensichtlich in

Korrektheit: $w \in \texttt{SUBSET SUM} \Leftrightarrow f(w) \in L$, offensichtlich, selbes Problem mit lediglich anders notierter Eingabe

2. Alternative Die Funktion f startet TM für L

Berechenbarkeit: Identität, offensichtlich in Polynomialzeit

Korrektheit: $w \in \texttt{SUBSET SUM} \Leftrightarrow f(w) \in L$ offensichtlich, selbes Problem