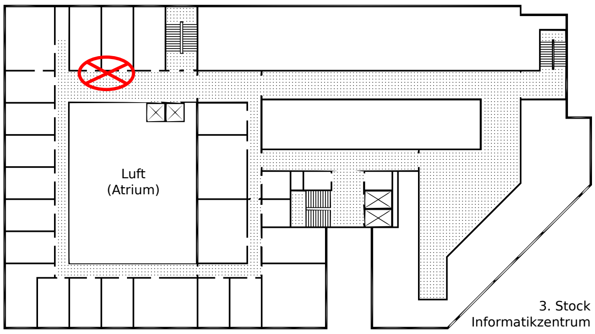
Algorithmen und Datenstrukturen 2 — **SoSe 2023**
Prof. Dr. Sándor Fekete — **Abgabe:** 10.05.2023
Dr. Phillip Keldenich — **Rückgabe:** ab 16.05.2023

Hausaufgabenblatt 1

Abgabe der Lösungen bis zum 10.05.2023 um 15:00 Uhr im Hausaufgabenschrank bei Raum IZ 337 (siehe Skizze rechts). Es werden nur mit einem dokumentenechten Stift (kein Rot!) geschriebene Lösungen gewertet. **Bitte die Blätter zusammenheften und vorne deutlich mit beiden Namen, Matrikel- Übungs- und Gruppennummer versehen!**



Hausaufgabe 1 (Fractional Knapsack): **(5+5 Punkte)**

In dieser Aufgabe betrachten wir Fractional Knapsack.

a) Sei $Z = 28$, und seien die acht Objekte mit folgenden Werten gegeben:

i	1	2	3	4	5	6	7	8
z_i	5	7	10	4	8	5	9	6
p_i	2	7	13	8	6	4	4	1

Wende den Greedy-Algorithmus für Fractional Knapsack aus der Vorlesung auf diese Instanz an. Gib in jeder Iteration den aktuellen Gegenstand, den Anteil x_i zu dem er gepackt wird, sowie den Gesamtzustand (aktueller Gesamtwert und aktuelles Gesamtgewicht) an.

b) Angenommen jedes Objekt steht beliebig oft zur Verfügung, d.h., für n Objekte $(z_1, p_1), \dots (z_n, p_n)$ sind nicht-negative Zahlen $x_1, \dots, x_n \in \mathbb{R}$ gesucht, sodass $\sum_{i=1}^{n} x_i z_i \leq Z$ und $\sum_{i=1}^{n} x_i p_i$ maximal.

Gib einen Algorithmus an, der das Problem optimal in Zeit $O(n)$ löst (d.h. insbesondere, dass er ohne Sortieren auskommt). Begründe außerdem kurz die Korrektheit Deines Algorithmus und die Optimalität der Lösung.

Hausaufgabe 2 (Maximum Knapsack): **(5+5 Punkte)**

In dieser Aufgabe betrachten wir Maximum Knapsack, also das Problem bei dem wir jedes gegebene Objekte nur ganz oder gar nicht nehmen können. Um dieses Problem zu lösen, wandeln wir den Algorithmus für Fractional Knapsack aus der Vorlesung folgendermaßen ab: Wir sortieren die Objekte aufsteigend nach $\frac{z_i}{p_i}$. Dann gehen wir die Objekte der Sortierung entsprechend durch und packen ein Objekt in den Rucksack, falls es hineinpasst. Eine formale Beschreibung ist in Algorithmus 1 zu sehen. Wir nennen diesen Algorithmus Greedy$_0$.

function $\textsc{Greedy}_0(z_1, \ldots, z_n, Z, p_1, \ldots, p_n)$
  Sortiere Objekte nach $\frac{z_i}{p_i}$ aufsteigend; dies ergibt Permutation $\pi(1), \ldots, \pi(n)$.
  for $j := 1$ **to** n **do**
    if $\left(\sum_{i=1}^{j-1} x_{\pi(i)} z_{\pi(i)} + z_{\pi(j)} \leq Z\right)$ **then**
      $x_{\pi(j)} := 1$
    else
      $x_{\pi(j)} := 0$
  return $x_1, \ldots, x_n$

Algorithmus 1: $\textsc{Greedy}_0$

a) Sei $Z = 14$, und seien die fünf Objekte mit folgenden Werten gegeben:

i	1	2	3	4	5
z_i	5	8	7	4	6
p_i	1	8	6	5	7

Wende $\textsc{Greedy}_0$ auf diese Instanz an. Entscheide in jeder Iteration, ob der aktuelle Gegenstand gepackt wird und gib den Gesamtzustand (aktueller Gesamtwert und aktuelles Gesamtgewicht) an. Ist die erhaltene Lösung optimal? Begründe Deine Antwort!

b) Zeige: Die ganzzahlige Lösung P_G von $\textsc{Greedy}_0$ kann beliebig weit von einer optimalen Lösung P_{OPT} (für das ganzzahlige Maximum Knapsack-Problem) entfernt sein. Mit anderen Worten: Zeige, dass es für jedes $c \in (0, 1]$ eine Instanz mit $0 < \frac{P_G}{P_{OPT}} < c$ gibt.
Hinweis: Es gibt geeignete Instanzen mit sehr kleinem n.

Lösung HA1

1. a)

$\frac{z_i}{p_i}$	2,5	1	0,77	0,5	1,3	1,25	2,25	6
RF	7	3	2	1	5	4	6	8

j	$\pi(j)$	$x_{\pi(j)}$	$z_{\pi(j)}$	$\sum_{i=1}^{j} x_{\pi(i)} z_{\pi(i)}$	$\sum_{i=1}^{j} x_{\pi(i)} p_{\pi(i)}$	$Z - \sum_{i=1}^{j} x_{\pi(i)} z_{\pi(i)}$
1	4	1	4	4	8	24
2	3	1	10	14	21	14
3	2	1	7	21	28	7
4	6	1	5	26	32	2
5	5	$2/8$	8	28	33,5	0

b) Durchlaufe alle Elemente und suche dabei das Element mit dem kleinsten Quotienten z_k/p_k

Setze $x_k = \frac{Z}{z_k}$ (nehmen Element k $\frac{Z}{z_k}$ mal)

<u>Korrektheit</u>: (Diese Beweise laufen normalerweise mit Widerspruch.)

Angenommen, es gibt eine bessere Lösung d.h. es wird ein Object m zu einem Anteil >0 genutzt.

Nehmen wir dieses Object m heraus und ersetzen es durch unser Obj. kann sich die Lösung nicht verschlechtern, da unser Obj. das beste Verhältnis hat.

Folglich ist unsere Lösung optimal.

```
min = z1/p1                 O(1)  ┐
ind = 1                     O(1)  │
for i = 2 to n                    │
    if zi/pi < min          O(1) ┐│
        min ← zi/pi         O(1) ├ O(n-1)   ├ O(n)
        ind ← i             O(1) ┘│
x_ind = Z/z_ind             O(1)  ┘
```

②

a) $Z = 14$

Obj	1	2	3	4	5
Z_i	5	8	7	4	6
P_i	1	8	6	5	7
Z_i/P_i	5	1	1,16	0,8	0,86
RF	5	3	4	1	2

j	$\pi(j)$	$Z_{\pi(j)}$	$x_{\pi(j)}$	$\sum_{i=1}^{j} x_{\pi(i)} \cdot Z_{\pi(i)}$	$Z - \sum_{i=1}^{j} x_{\pi(i)} \cdot Z_{\pi(i)}$	$\sum_{i=1}^{j} x_{\pi(i)} P_{\pi(i)}$
1	4	4	1	4	10	5
2	5	6	1	10	4	12
3	2	8	0	10	4	12
4	3	7	0	10	4	12
5	1	5	0	10	4	12

nicht optimal!

bessere Lösung: Obj 2+5

Wert = 15

b) $\forall c \in (0,1]$ existieren Instanzen, sodass

$$0 < \frac{P_G}{P_{opt}} < c$$

$Z = N$

Obj	1	2
z_i	1	N
p_i	1	$N-1$
z_i/p_i	1	$N/(N-1) > 1$
RF	1	2

Sei $Z = N$ für bel. $N \in \mathbb{N}$ und wähle zwei Objekte 1, 2.

Da $\frac{z_1}{p_1} < \frac{z_2}{p_2}$ wird Greedy$_0$ Obj 1 packen und erhält einen Wert von $P_G = 1$.

Die optimale Lösung besteht aus Obj 2. mit $P_{opt} = N-1$

Wir erhalten $\frac{P_G}{P_{OPT}} = \frac{1}{N-1}$ Für $N > \frac{1}{c}$ ist der Quotient kleiner c.