

## Homework Assignment #8

### Approximation Algorithms (Winter Semester 2023/24)

#### Exercise 1 – Greedy for the KNAPSACK problem

Consider the following greedy approach for the KNAPSACK problem: Iterate over the objects in descending order according to the ratio between profit and size, i. e.,  $\frac{\text{profit}(a_1)}{\text{size}(a_1)} \geq \dots \geq \frac{\text{profit}(a_n)}{\text{size}(a_n)}$ . Add the current object to the knapsack if there is enough space left.

Show that this approach has an arbitrarily bad approximation ratio.

[5 points]

#### Exercise 2 – 1/2-Approximation for the KNAPSACK problem

Consider the following algorithm for the KNAPSACK problem: Let  $a_1, \dots, a_n$  be the objects in descending order according to the ratio between profit and size. Assume that each individual object fits in the knapsack. Now determine a smallest index  $k$  such that the total size of the first  $k$  objects exceeds the capacity of the knapsack. Finally, return the better of the two solutions  $\{a_1, \dots, a_{k-1}\}$  and  $\{a_k\}$ .

Show that this algorithm is a 1/2-approximation.

[7 points]

#### Exercise 3 – MINIMUM MAKESPAN SCHEDULING on Two Machines

Consider the following problem: Given  $n$  jobs with runtimes  $t_1, \dots, t_n \in \mathbb{N}$ , distribute them over two machines such that the total makespan is minimal. More formally, we search for a partitioning of  $T := \{t_1, \dots, t_n\}$  into two sets  $M_1$  and  $M_2$  such that  $t(M_1, M_2) := \max\{\sum M_1, \sum M_2\}$  is minimized.

Develop an FPTAS for this problem.

[8 points]

*Suggestion:* First, develop an exact algorithm with pseudo-polynomial runtime. Then, scale and round accordingly.