Please read carefully and start with answering those questions that you are sure you can solve. Write clearly, always state which exercise you are answering when using extra sheets, and indicate by crossing out if you want parts of your answer to be disregarded. Any ambiguity or unclarity will result in zero points, or in the worst of several solutions being graded. If you are uncertain of a formal statement or proof, write it down informally rather than giving no solution.
Note: you can write all solutions on the exam sheet, there is enough space reserved after each exercise. If you prefer or need extra space, you can also use exam paper. The exam has 14 pages and a total of 100 points.

Best Wishes!

Question 1 (17 points)
Which of the following statements are true given the assumptions in the following parts? Give a short reason for your opinion.

a) (4 points) Assume that A is a Monte-Carlo Algorithm for problem T.
Hint: *'Monte Carlo' were those randomized algorithms with guaranteed bound on the running time.*

1) Then, A can be converted into an according Las-Vegas Algorithm.

2) Then, A cannot be the algorithm that guesses a variable assignment for a given $3SAT$-formula, checks if the assignment is fulfilling, and repeats in case it is not.

b) (6 points) Assume that A is a PTAS for maximization problem T, but it is not an FPTAS. Assume $\varepsilon \in (0,1)$.

1) Then, A is a family of algorithms such that for every ε, $A_\varepsilon \in A$ produces a solution ALG_ε to T s.t. $(1-\varepsilon)OPT \leq ALG$, where OPT the value of an optimal solution for T.

2) Then, the running time of any $A_\varepsilon \in A$ is exponential in the input size of the according instance of problem T.

3) Then, the running time $R(A_\varepsilon)$ of some $A_\varepsilon \in A$ satisfies $R(A_\varepsilon) \in \Omega(\frac{1}{\varepsilon^{100}})$.

c) (4 points) Assume that ALG is a (possibly randomized) *online* algorithm for maximization problem T.

1) Then, for the (expected) tight competitive ratios of ALG against an oblivious (α_{ob}) and against an adaptive adversary (α_{ad}), it holds (note that always $\alpha \geq 1$ in our definition)

$$\frac{1}{\alpha_{ad}} \leq \frac{1}{\alpha_{ob}} \leq 1 \ .$$

2) Then, for α_{ob} defined as above, there exists an offline algorithm for T which outputs a solution of value at least $\frac{1}{\alpha_{ob}} OPT$.

d) (3 points) Assume the randomized global MinCut algorithm from the lecture (the one that chooses a random edge and contracts it) is run on a graph $G = (V, E)$ with $|E| = n$ for which it is previously known that the minimum cut in G has no more than k edges. Then, the algorithm's probability of success it at least

$$P \geq \prod_{i=0}^{n-(k+1)} \left(1 - \frac{k}{n-i}\right)$$

Question 2 (16 points)
Name two algorithmic paradigms/modeling techniques from the lecture that should be useful to solve the following problems optimally. **For one of them, give a short argument on why you expect it to work. For the other, also give an algorithm or a representation that solves the problem** (e.g., a greedy procedure, a formula for DynProg, a Flow Network). You do **not** need to state a proof of correctness.

Hint: *those techniques include Greedy, Dynamic Programming, Informed Search, Network Flows, Linear Programs, Integer Linear Programs, Randomized Algorithms, Online Algorithms.*

a) (8 points) The Coin Problem: assume your country produces coins of values $\{1, 2, 5\}$. Given a natural number I, what is the minimum number of coins that can be used to pay I? (Make sure that at least one of your solutions still works if the values of the two larger coins are exchanged for any other natural numbers, i.e. any set $\{1, n_2, n_3\}$).

b) (8 points) The Market Problem: you can present your products at k different stores $S = \{s_1, \dots, s_k\}$. You have n different products $P = \{p_1, \dots, p_n\}$ which each have a revenue value $v_i \in \mathbb{R}_+$, $i \in \{1, \dots, n\}$, that will be obtained in each store that sells them. However, each store s_j can only sell a certain, given subset P_j of all products. Also, each product has limited availability captured by a_i, $i \in \{1, \dots, n\}$, which means that p_i can go to at most a_i stores.
Maximize the overall revenue obtained by assigning each store a subset A_j of the products, i.e. maximize $\sum_{j\in\{1,\dots,k\}} \sum_{i \in A_j} v_i$.

Question 3 (8 points)
The Subset-Sum problem is NP-complete. Use this to show that the Knapsack Problem is NP-complete (recall the definition of NP-completeness and ensure to show both parts).

Subset-Sum: given natural numbers $N = \{n_1, \ldots, n_m\}$ and $T \in \mathbb{N}$, does there exist a subset $S \subseteq N$ such that $\sum_{n_i \in S} n_i = T$?

Knapsack: given items $\{1, \ldots, k\}$ with costs $c_i \in \mathbb{R}_+$ and values $v_i \in \mathbb{R}_+$ for all $i \in \{1, \ldots, k\}$, as well as a knapsack size B and target value T: is there a subset $S \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in S} v_i \geq T$ and $\sum_{i \in S} c_i \leq B$?

Question 4 (8 points)

a) (3 points) Show $4n^5 \log n \notin \Omega(n^6)$.

b) (2 points) What does it mean for a problem P to have Integrality Gap at most 4?

c) (3 points) What requirements should a heuristic satisfy in the A^*-algorithm?

Question 5 (8 points)
Consider Max-4-SAT in CNF with only one occurrence of the same variable per clause.
Hint: *Max-4-SAT is the problem of finding an assignment for the variables that satisfies as many clauses as possible. CNF is short for conjunctive normal form.*

a) (2 points) Give a simple, randomized algorithm to find an assignment that yields, in expectation, a constant-factor approximation to the optimum.
Hint: *use the one from the lecture for other versions of this problem.*

b) (4 points) Prove its expected approximation ratio.

c) (2 points) State the same algorithm's approximation ratio for the problem Max-k-SAT, with $k \geq 2$.

Question 6 (14 points)

a) (6 points) Choose one of the two following statements, and give a reduction or algorithm to support your choice: Deciding whether a given undirected graph G has a 2-vertex-coloring is (a) NP-hard, or (b) in P. You do not have to prove the correctness of your reduction or algorithm.

b) (2 points) Give an example of a graph that is not 2-vertex-colorable.

c) (6 points) Characterize the class C of all 2-vertex-colorable graphs. Prove that indeed, $(G \in C) \Leftrightarrow$ (G is 2-vertex-colorable).

Hint: *In case you proved NP-hardness above, use the NP-hard problem you reduced to 2-coloring to identify the structural requirements for such a 2-coloring. If you gave a polynomial algorithm for the problem, identify when exactly your algorithm fails. If you did neither, work from your solution to b).*

Question 7 (6 points)
Consider the following variant of a flow network: we get a graph $G = (V, E)$, two nodes $s, t \in V$, but instead of a capacity for each edge, we get a non-negative demand given by a function $d : E \to \mathbb{R}^+$, stating the *minimum* amount of flow to send through each edge. The *min-s-t-flow* problem looks to send a flow f of minimal value from s to t such that the flow $f(e) \geq d(e)$ for each $e \in E$.

Prove or disprove: the value of a minimal s-t flow equals the value of a maximum s-t-cut.

Question 8 (8 points)

Prove that the k-Independent Set Problem is not in APX. The problem is as follows: given undirected graph $G = (V, E)$ and number k, decide whether there exists an independent set of size k in G. An independent set is defined as a subset $S \subseteq V$ such that for any $x \in S$, $y \in S$, $(x, y) \notin E$.

Hint: *use the inapproximability-result for k-Clique mentioned in the lecture (a k-clique is a subset $C \subseteq V$ of k vertices such that all edges (x, y) with $x, y \in C$ are in E).*

Question 9 (15 points)
Consider a network $G = (V, E)$ of streets E between villages V in a rural area. You want to transport goods along different routes from your factory $s \in V$ to the harbor $t \in V$, and need to send k trucks to transport everything. There are three acceptable routes P_1, P_2, P_3 that you know lead to the harbor and you can use. Each such route corresponds to a simple s-t-path in G, i.e. $P_i \subseteq E$ for all $i \in \{1, 2, 3\}$ (and routes might share certain streets).

Still, streets are bad and the area is quite dangerous, so road blocks are a frequent problem. You therefore aim to minimize *congestion*, i.e. the maximum number of trucks using the same street. I.e., assign each of the k trucks one of the routes to minimize the following:

$$\max_{e \in E} (\text{number of trucks using street } e)$$

a) (6 points) State an ILP for the above problem and give the according LP-relaxation.
Hint: *as always, start with defining variables that represent the decisions to make in this problem. For the maximum, use an extra variable m that you bound from below accordingly.*

b) (4 points) State a constant bound on the maximal integrality gap of your ILP. Give a 1-sentence-argument on why this bound holds.

c) (5 points) Give a rounding scheme that, given an optimal solution x^* to the LP, computes an integer solution x to the ILP that is a constant-factor approximation to the optimum LP value.

Hint: *Note that this is even possible without stating the full ILP.*