APMA 1210 – FINAL EXAM – STUDY GUIDE

The Final Exam takes place on

Wed, December 14, 2022, 2−5 pm in 117 Macmillan Hall

It will be an in-person exam and **NO** books/notes/calculators/cheat sheets will be allowed. The final counts for 35% of your grade, and covers the whole course. There will be emphasis on the material after Midterm 1, and equal emphasis on the Midterm 2 and post-Midterm 2 materials.

This is the study guide for the exam, and is just meant to be a *guide* to help you study, just so we're on the same place in terms of expectations. For a more thorough study experience, look at the lecture notes, homework, midterms, and (possibly) practice exam. I will **NOT** ask you anything in the book that I haven't covered in lecture/homework/practice exam, so no need to read the book.

Format: There are 7 questions on the exam, all of them free response, no multiple choice. One of them will be a simplex method problem, and the rest from the material below. There will be a word problem, and there might be a proof.

Lecture 1: Intro to Operations Research

You can ignore this lecture, it's just a historical overview.

Date: Wednesday, December 14, 2022.

Lecture 2: Intro to Mathematical Programming

- Find the decision variables, objective function, and constraints of a word problem, and write down the complete optimization problem. Great practice questions are the examples from lecture, as well as the homework and the coffee bean midterm problem.
- Write an LP problem in matrix form
- Also check out the Integer Linear Programming Problem from that lecture

Lecture 3: Linear Programming

- Put an LP problem in standard form. There are two rules: The Sign Rule and the Slack Variables Rule. I might ask you to write it in matrix form.
- Conceptually understand why the max/min has to be at a vertex, using the level curves (like $z = 10$ and $z = 15$ in the lecture example)
- Ignore the cookie problem

Lecture 4: Geometry of Feasible Regions and Convexity

- You don't need to memorize the definitions of polyhedron, hyperplane, and half-space, but just know what they are. For example, when I write $a^T x = b$, know that this represents a hyperplane.
- Remember that a is perpendicular to the hyperplane $a^T x = b$

- Know the definition of a line segment between x and y
- Know the definition of a convex set
- Show that a half-space $a^T x \leq b$ is convex (I would give you the definition of half-space)
- Know the definitions of convex combination and convex hull
- No need to know the alternative definition of a polyhedron
- You don't need to memorize the definitions 1 and 2 of a vertex, but please understand what they mean. For example, I would provide you the definition of an extreme point.
- Remember the proof with convexity on your homework.

Lecture 5: Extreme Points of Feasible Regions + Simplex Algorithm (I)

- Define active, basic solution, and basic feasible solution
- It's good to check out the examples that follow the definition, especially the pyramid one
- You can ignore the section on "Existence of Vertices"
- Ignore the proof in the section "Optimality and Extreme Points," but know the results, especially the "Glorious Result"
- The simplex method is done in detail in the following lecture

Lecture 6: Simplex Algorithm (II)

- I will definitely ask you an example of the simplex method, but I will probably just do something like 2 variables and 2 constraints, to make the algebra a bit more manageable
- On the midterm, I asked you to do one step, but this time there might be two (or more) steps
- Make sure to understand Example 1 in detail
- For the very last step, you don't need to write the constraints in terms of z_1 and z_2, since we know the point is already optimal. That said, you need to write the objective function in terms of z_1 and z_2
- Know how the higher-dimensional example presented in lecture or the homework works. If I ask a higher-dimensional example, it would be something like "Perform one step of the simplex method"
- Ignore the section "MATLAB Implementation." As a general rule, for the exams you can ignore the coding parts, that's more for the homework.

Lecture 7: Simplex Algorithm (III)

- Define local min (see homework), global min (see homework), and convex function
- Show that a local min of a convex function is a global min (see homework); I would give you the hints in this case
- Show that $z = c^T x$ is convex

- Understand the trick with $z_1 + \cdots + z_m$ to obtain a starting vertex. Here I'll make sure that the b_i are ≥ 0, to prevent any confusion

- Ignore the sections on "Degeneracy" and "Efficiency"

Lecture 8: LP Duality (I)

- Ignore the section on "Motivation" if you want, although I think that's pretty useful to understand why we have a min

- Define and find the dual problem to an LP problem

- I would memorize the sign rules just in case, since you'll need them later on, at least the ones with $\leq$ and $=$

- Show that the dual of the dual is the primal

- Know the statements of the weak and strong duality theorems, but you don't need to know the proofs. Know how to prove the Corollary and the Co-Corollary

Lecture 9: LP Duality (II)

- You don't need to memorize the table with the possible scenarios, but know how to justify the ones with the X mark

- Know how to set up the LP problem for the farmer, and find the dual problem. No worries if you don't understand the "Interpretation part"

- Know how to set up the primal problem with slack and the dual problem with excess

- Know complementary slackness and the Example 2 that follows. For the proof, I would give you a hint like "Consider $y^T Ax$." No need to really understand the "Dual Problem" part that follows

Lecture 10: Sensitivity Analysis

- Define shadow price
- Thoroughly study the example from lecture
- Know that there's a relation between shadow prices and dual LP problems
- Define reduced cost and understand the relation with dual LP problems
- Ignore the section "Other Sensitivity Issues"
- Homework 4: Problems 1 and 2, although remember that there's no MATLAB allowed

Lecture 12: Game Theory

- Find the gains matrix of a game, like here Rock/Paper/Scissors
- Understand what a mixed strategy is
- Define expected payoff (for player 1)
- *Thoroughly* understand the Pokémon example and how it leads to a LP problem
- In particular, write down an LP problem for both Pikachu's and Charizard's strategy, with matrices and without matrices (where you just write down everything explicitly)

- Understand what this has to do with Dual LP problems
- Ignore the section on MATLAB implementation
- Homework 4: Problem 3

Lecture 13 + 14: Network Problems

- Define: Graph, Directed Graph
- You don't need to understand the Google Graph
- Find the Adjacency Matrix and the Oriented Incidence Matrix of a graph.
- *Thoroughly* understand the coffee problem and how it leads to an LP problem, as well as all the vocabulary associated to it like capacity constraints and conservation of flow; this is continued in the next lecture
- Understand all the variations talked about in lecture: Transportation Problem, Assignment Problem, Max Flow, Shortest Path. I could ask you a word problem with those
- Define: Connected, Disconnected, Cycle, Tree, Degree, Leaf, Branch
- Homework 5: Problems 1, 2, 3, 4 and Midterm 2: Problem 1

Lecture 15: Network Simplex Algorithm

- Show that every finite tree has a leaf
- Show that a tree with n vertices has $n-1$ edges
- Know the two other facts about trees, but you don't need to prove them
- Understand the correspondence between Spanning Trees and Invertible 3×3 sub-matrices in the example in lecture, it is very interesting
- Find a tree solution (initial vertex) of a graph, you need this to start the network simplex algorithm
- You don't need to know the technical definition of a tree solution
- Implement the network simplex algorithm to find an optimal tree; this is continued in the next lecture. Notice this involves:
  (1) Finding an initial tree, using the algorithm above
  (2) Checking if the initial tree is optimal, using the reduced cost
  (3) If not, adding/removing an edge
  (4) Checking if the new tree is optimal, using the reduced cost
- Homework 6: Problem 1 and Midterm 2: Problem 2

Lecture 16: Max Flow/Min Cut

- Solve a max flow problem using the method of residual graphs
- Define cut

- Notice a cut doesn't have to be a line, and for the value of the cut you only consider edges pointing out of the $s-$region
- Solve a max flow problem using min cuts
- Understand that max flow = min cuts
- Show that a graph with n intermediate cities has 2^n cuts.
- Homework 6: Problems 2, 3, 4 and Midterm 2: Problem 3

Lecture 17: Dynamic Programming

- Understand how dynamic programming works, and in particular how you use your friend f to solve the problem implicitly.
- You can ignore the section "Ordering on vertices," in the problem that I would give the ordering will be clear
- Solve a problem using dynamic programming, you would need to draw all the tables and give me the shortest path and total cost
- Ignore the section "Distance between Words"
- Homework 7: Problem 1 and Midterm 2: Problem 4

Lecture 18: Integer Programming (I)

- Define: Integer program, Mixed Program, Binary Variable
- Review the Multiple choice example from lecture
- Know how to represent event conditions. For example, to say $x_2 \to x_1$, you would use $x_2 \leq x_1$

- Review the Airline Example from lecture
- Homework 7: Problem 2

Lecture 20: Integer Programming (II)

- Express "or" conditions by introducing extra binary variables
- Similarly, deal with the case where our variables take on more than 2 values, like $x_1 \in \{2, 5, 8\}$
- Definitely review the hospital problem, that one is quite important and illustrates all the important techniques
- You don't need to know the trick where you sum the x_{ij} over all i, but do notice how the number of constraints goes down dramatically
- Review the section "Geometry of IP Problems," it's a good illustration of how not to solve IP problems
- Know what a relaxed LP is and the fact that IP $\leq$ LP
- Once again, it's good to review Problem 2 in Homework 7

Lecture 21: Branch-and-Bound

- Please review the branch-and-bound method! Even though I didn't ask that on the homework, it's fair game for the exam, and a really neat way of solving integer programming problems
- You can ignore the sections "Remarks" and "Strong Formulation"

Lecture 22: Non-Linear Programming (I)

- You don't need to memorize all the problems that arise with nonlinear programming, bust just know that the max/min might not be at a vertex
- Set up the two fun applications problems (Peyamazon and Ice Cream)
- Once again, know the definition of convex set and convex/concave functions
- Once again, convexity is useful because a local min of a convex function is a global min
- Know the definition of critical point and the second derivative test from calculus

Lecture 23: Non-Linear Programming (II)

- Know the definition of critical point in several variables and how to find it
- Calculate the Hessian $D^2 f$
- Know the True second derivative test
- Find the local max/min/saddle points of a function in several variables
- Show that f is convex. You need to know both methods, by calculating $D^2 f$ and by calculating the leading principal minors D_2 and D_1 (and D_3 if you're in 3 dimensions)
- Show that f is concave. You can either show that $-f$ is convex or show $(-1)^n D_n > 0$ for all n

- Find the global min/max of f. This amounts to finding the critical point and showing that f is convex/concave
- Check out the 3 application problems: Clustering, Linear Separators, and Regression
- Also don't forget about the neat proof problem in Homework 7. I could ask you to re-prove this, with all the hints
- Homework 7: Problem 3 and 4

Lecture 24: Gradient Descent

- You don't need to know how we obtained the function f in the introduction, and you don't need to understand the motivation
- Use cyclic ascent to find an approximate local max of a function. I would tell you how many steps to use, and the algebra will be a bit more manageable. Try for example doing steps 0 to 3 in the example in lecture by hand
- Define ∇f, the directional derivative of f in the direction $\langle 3, 4 \rangle$ and find the direction of the greatest rate of increase/decrease of f at a point.
- Use steepest ascent to find an approximate local max of a function. I would tell you how many steps to use, and the algebra will be much more manageable than the example in lecture.