LEARNING COMMONS

FRANKLIN UNIVERSITY PROFICIENCY EXAM (FUPE)
STUDY GUIDE

Course Title: MATH 320: Discrete Mathematics

Recommended Textbook(s): https://www.franklin.edu/current-students/academic-resources/textbooks

Number & Type of Questions: 90 – Multiple Choice and true/false

Permitted Materials: Appendix with tables of logical equivalencies and logical inferences (linked in the test)

Time Limit: 240 minutes (4 hour)

Minimum Passing Score: 75%

Format varies

Outline of the Topics Covered:

Course Description

This course introduces students to fundamental algebraic, logical, and combinational concepts in mathematics that are needed in upper-division computer science courses. Topics include sets, mappings, and relations; elementary counting principles; proof techniques with an emphasis on mathematical induction; graphs and directed graphs; Boolean algebras; recursion; and applications to computer science.

Knowledge & Skills Required

1. Construct truth tables for compound propositions involving conjunction, disjunction, exclusive or, implication, and negation (and various combinations thereof). (As in Section 1.1, #s 31-35.)
2. Use truth tables to prove or disprove that some logical propositions are equivalent. (Section 1.3)
3. Give the truth-value of some propositions involving quantifiers. (As in Section 1.4, #s 11-16.)
4. Find the union, intersection, difference, symmetric difference, complement, and power set of various sets. (Sections 2.1 and 2.2)
5. Determine whether or not a function is one-to-one, onto, and/or a one-to-one correspondence. Be able to give reasons for your answers. (Section 2.3)
6. Evaluate some summation expressions. (As in Section 2.4, #s 29-34.)
7. Find a specific term in a given sequence. (As in Section 2.4, #s 1-4.)
8. Apply the definition to show that a given function is “big-O” of another function, specifying values for the constants C and k. (Section 3.2)
9. Determine how much time an algorithm will take or how large a problem can be solved in one second. (As in Section 3.3, #s 15, 18, 19.)
10. Determine whether or not two matrices can be multiplied, and if it is possible find their product. (Section 2.6)
11. Determine whether or not two matrices are inverses of one another. (Section 2.6)
12. Find the join, the meet, and the Boolean product of two zero-one matrices. (Section 2.6)
13. Identify the propositions in a worded argument and state what rule of inference is used. (As in Section 1.6, #s 3, 4, 7, 8, 9, 10.)
14. Identical a proof as being a direct proof, an indirect proof, or a proof by contradiction. (Section 1.7)
15. Prove some propositions using direct proof, indirect proof, or proof by contradiction. (As in Section 1.7, #s 1, 3, 9, 10, 11, 12, 17, 18, 19, 20, 21, 26, 27, and in Section 1.8, #s 3, 4, 7.)
16. Use mathematical induction to prove some propositions. (As in Section 5.1, #s 3, 6, 7, 8, 10, 11, 15, 19, 20, 21, 32, 33.)
17. Use a recursive definition to find particular functional values. (As in Section 5.3, #s 1, 2, 3, 4.)
18. Give recursive definitions for sequences or functions defined otherwise. (As in Section 5.3, #s 711, 23, 24a) b), 25.)
19. Describe some recursive algorithms. (As in Section 5.4, #s 7-12, 29.)
20. Compare the merits of recursive versus iterative algorithms; that is, what are the advantages and disadvantages of using one type of algorithm rather than the other? (Section 5.4)
21. Apply the “sum rule,” the “product rule,” the “pigeonhole principle,” and the “generalized pigeonhole principle” to problems. (Sections 6.1 and 6.2)
22. Distinguish between situations that involve permutations of objects and situations that involve combinations of objects and solve problems involving these concepts. (Section 5.3)

23. Recognize the various notations used to indicate permutations and combinations and do the computations they require. (Section 6.3)
24. Write a binomial expansion or give the coefficient of a particular term of a binomial expansion. (As in Section 6.4, #s 2, 4, 6, 7, 8, 9.)
25. Write several terms of a sequence for which you are given the recurrence relation and initial conditions. (As in Section 2.4, #9.)
26. Find a recurrence relation satisfied by a given sequence. (As in Section 2.4, #14.)
27. Find the solution to a recurrence relation for which you are also given the initial conditions. (As in Section 2.4, #17.)
28. Set up a recurrence relation, write an explicit formula, and find a particular term of the sequence for an application problem. (As in Section 2.4, #s 18, 19, 20, 21, 22; section 8.1, #s 11, 12.)
29. Find a recurrence relation for a bit-string problem. (As in Section 8.1, #s 7, 8, 9.)
30. Find the solution of a linear homogeneous relation with constant coefficients. (As in Section 8.2, #s 3, 4, 12-15.)
31. Write recurrence relations and then solve application problems. (As in Section 8.2, #s 5-9.)
32. Write the adjacency matrix for a directed or an undirected graph. (Section 10.3)
33. Give the isomorphism for a pair of graphs or explain why the pair is not isomorphic. (Section 10.3)
34. Identify whether or not a graph is connected. (Section 10.4)
35. Find the dual of a Boolean expression. (Section 12.1)
36. Find the sum-of-products expansion of a Boolean expression. (Section 12.2)
37. Find the output of a given circuit or construct a circuit that produces a given output. (Section 12.3)
38. Use a Karnaugh map to find a minimal sum-of-products expansion of a Boolean function. (Section 12.4)
39. Use the Quine-McCluskey method to find a minimal sum-of-products expansion for a Boolean function. (Section 12.4)
40. Convert binary numbers to hexadecimal and to decimal numbers; convert hexadecimal numbers to binary and to decimal numbers; convert decimal numbers to binary and hexadecimal numbers.
41. Covert numbers in fixed-bit two’s complement representation to decimal form; convert decimal numbers to fixed-bit two’s complement representation.
42. Do some binary, some hexadecimal, and some two’s complement addition problems, watching for overflow situations in the two’s complement problems.