

הודעה לסטודנטים לתואר שני במדעי המחשב (M.Sc.)

הנדון: בחינת הכשר

ביום ו' ה-13 באפריל 2018 תיערך בחינת הכשר לתואר השני, בשעה 9:00 בבוקר.

ההרשמה לבחינה תיערך ע"י פנייה במייל לרכזת התוכנית, תמר אשל tamar.eshel@idc.ac.il

לתשומת ליבכם: ניתן להירשם לבחינה עד לתאריך 30/3/18 בלבד.

**** אנא ציינו במייל את שמכם המלא ומספר זהות ****

לתשומת לבכם מספר הנחיות

- ❖ כל סטודנט לתואר שני חייב עד סוף שנת הלימודים השנייה להגיש הצעה למחקר או פרויקט גמר או לעבור בחינת הכשר. אם עברתם כבר את שנת הלימודים הראשונה שלכם ועדיין לא התקדמתם משמעותית בגיבוש נושא לתזה או פרויקט, עליכם לגשת לבחינת ההכשר הקרובה כדי למנוע עליה לוועדה אקדמית בקיץ. האפשרות להגיש הצעה למחקר או פרויקט תישאר פתוחה בפניכם בכל מקרה במהלך השנה הקרובה. במקרה של ספק או שאלה, אתם מוזמנים לפנות לייעוץ אצל ראש התכנית, ד"ר אילן גרונאו ilan.gronau@idc.ac.il
- ❖ סטודנט שלא מתקדם במסלול המחקרי (תזה או פרויקט גמר), מחויב בנוסף לבחינת ההכשר, ללמוד שלושה קורסי בחירה נוספים (סה"כ 36 נ"ז קורסי בחירה) על מנת לסיים חובותיו האקדמיות.
- ❖ הציון בבחינת ההכשר הוא עובר/לא עובר ואינו משוקלל בממוצע הכללי של התואר.
- ❖ ציון עובר בבחינת ההכשר הוא תנאי מחייב לזכאות בתואר עבור סטודנטים שלא השלימו תזה או פרויקט גמר.
- ❖ הבחינה תיערך בכתב.
- ❖ בחינת ההכשר מכסה מגוון נושאי ליבה במדעי המחשב במטרה לבדוק בקיאות כללית במושגים השונים והיכרות עם טכניקות שונות לפתרון בעיות. לא יידרש פתרון טכני של בעיות קונקרטיות.

מצורף בהמשך:

סילבוס עם נושאי הבחינה

פרוט הנושאים שיופיעו בבחינה:

List of topics for the M.Sc. qualification exam

Data Structures:

1. List, Queue, Stack - ADT implementation: pointers, array, cursors
2. The definition of Order: upper, lower and tight bounds.
3. Sorting:
 - QuickSort, MergeSort, BubbleSort
 - Lower bound for comparison sorts
 - Sorting in linear Time: Counting, Radix and Bucket Sort
4. Heap
 - Definition & operations
 - Heapsort
5. Hash Table
 - Open hash, Close hash – definition & operations
 - Perfect hash – only definition
 - Universal hash – only definition
6. Binary Search Trees
 - Definition & operations
 - Traversal: Inorder, Postorder, Preorder
 - One balance tree: AVL or Red-Black or 2-3 (to the student choice)

Textbook: *Introduction to Algorithms*, by T. Cormen, C. Leiserson, R. Rivest and C. Stein.

Operating Systems:

1. Computer and operating systems structure
 2. Processes and threads
 3. CPU scheduling
 4. Process Synchronization (mutual exclusion etc.)
 5. Deadlocks
 5. Memory management
 6. Paging and Virtual memory
- (No need to cover Input/Output; File systems)

Textbook: *Modern Operating System, Second Edition* by Andrew S. Tanenbaum
Publisher: Prentice Hall, 2001 (ISBN 0-13-031358-0). **Chapters 1,2,3,4.**

Digital Architecture:

1. Boolean algebra and combinational logic
 - Implementing Boolean function using gates
 - Simplifying Boolean functions using Karnaugh maps
 - Cost and delay measures of a design
 - Decoders and multiplexers
2. Fixed point arithmetic
 - Unsigned numbers
 - Representing fractions in fixed point
 - Two's complement numbers
 - Negating numbers
 - Sign extension
 - Double precision
 - Adders, fast adders & ALUs
3. Sequential logic
 - Master-slave Flip-flops & registers
 - Registers with write enables
 - Counters
 - State machines
4. From C to machine language
 - MIPS assembly language
 - Compilation, linking & loading of C programs (from C to machine language)
 - C procedure convention – Stack frame, transferring arguments, allocation of local variables
 - Recursion
5. Computer structure
 - Single-cycle MIPS implementation
 - Multi-cycle MIPS implementation
 - Exceptions and interrupts handling

Textbooks:

- 1) Lecture notes that can be downloaded from the Digital Architecture course web page
- 2) "Computer Organization & Design: The Hardware/Software Interface", by D. A. Patterson and J. L. Hennessy, Morgan Kaufman Publishers, Inc. 2nd edition, 1998, or 3rd edition, 2005.

Algorithms:

- Topological sort
- Euler and Hamiltonian paths
- Trees, characterizations of trees.
- DFS – Depth first search
- BFS – Breadth first search

- Shortest paths:
 - Weighted or non-weighted edges
 - Positive weights, non-negative cycles
 - Directed, undirected or directed acyclic (DAG) graphs.
 - Single source or all-pair shortest path.
- Minimum spanning tree, Kruskal's and Prim's algorithms.
- Network flow; finding max-flow and min-cut; matching in bipartite graph.
- Greedy algorithms, general structure, Huffman code.
- Dynamic programming.

Textbook: *Introduction to Algorithms*, by T. Cormen, C. Leiserson, R. Rivest and C. Stein. MIT Press. (1st edition or open univ. translation are also good.)

Computer Networks:

1. Computer networks and the internet
2. Application layer
3. Transport layer
4. Network layer
5. Link layer

Textbook: *Computer Networking: A Top Down Approach Featuring the Internet, third edition*, by J.F. Kurose and K.W. Ross, Publisher: Addison-Wesley Longman (ISBN: 0-321-26976-4). **Chapters 1,2,3,4,5.**

Automata and Formal Languages:

- Regular Languages:
 - Deterministic automata.
 - Non-deterministic automata, equivalence to deterministic model
 - Regular expressions
 - Closure of regular languages to various operations.
 - Non-regular languages, pumping lemma for regular languages.
 - Minimization of automata
- Context-free languages:
 - Context-free grammars
 - Pushdown automata
 - Pumping lemma for context-free languages.
 - Closure of context-free languages to various operations.

Textbooks:

Introduction to Theory of Computation - M. Sipser
 'אוטומטים ושפות פורמליות' – האוניברסיטה הפתוחה.

Computation and Complexity:

- Computability theory
(Chapters: 3-5)
 - Turing machines: Definition and the difference between TM and finite automata, pushdown automata, and a desktop computer.
 - Church-Turing thesis
 - Decidable and recognizable languages: definition and examples.
 - Undecidable languages: Halting problem and other examples.
 - Reducibility: definition and its utilization.
- Complexity theory
(Chapters: 7-8)
 - Time complexity: definition.
 - Definition and examples for each of the following complexity classes:
P, NP, NP-complete, NP-hard, EXP
 - NP-complete problems: CLIQUE, Hamiltonian Path, Vertex Cover, 3SAT
 - Polynomial reductions: definition and its utilization.
 - Space complexity:
PSAPACE, LOG-Space, NLOG-Space (definition and examples).

Textbook: Introduction to Theory of Computation - M. Sipser

Programming Languages:

1. Functional programming:
 - Lambda notation
 - Functions as first-class objects
 - Higher-order functions
2. Models of programming languages
 - Abstract data types
 - The substitution model
 - Applicative-order and normal-order evaluation
 - The environment model
 - Data-directed programming, object-oriented programming and message passing
3. Interpretation
4. Object-oriented principles
 - Quality and modularity
 - Classes and objects
 - Memory management
 - Genericity
 - Inheritance and multiple inheritance
5. Design by contract

Textbooks: (1) *Structure and Interpretation of Computer Programs*, by Gerald J. Sussman and Harold Abelson, 2nd edition. Chapters 1-4. Publisher: MIT Press, 1996. (2) *Object-Oriented Software Construction*, by Bertrand Meyer, 2nd edition. Chapters 1, 3-12, 14-16, 20-24. Publisher