

Pokrytí otázek testu GRE CS předměty OI

Jan Kybic

June 10, 2011

Sestaveno na základě dokumentu “GRE Computer Science Test Practice Book” z roku 2004¹ a dokumentu “Bakalářský a magisterský program Otevřená informatika” z 26.ledna 2011 2008.² Třetí verze dokumentu.

1 Témata obsažená v testu

Níže najdete témata jejichž znalost GRE CS test ověřuje. Zelený kód kurzívou (např. *LAG*) je kód předmětu, který příslušnou látku obsahuje. Oranžové kódy s otazníkem (např. *LAG?*) znamenají, že si nejsme jisti, zda předmět látku obsahuje, ale myslíme si, že by měl. Červený symbol \emptyset znamená, že téma pokryto není, nebo je pokryto jen částečně.

I. SOFTWARE SYSTEMS AND METHODOLOGY (40%)

A. Data organization

1. Data types. *PR1*, do *ALG* nepatří \emptyset
2. Data structures and implementation techniques. částečně *ALG* \emptyset

B. Program control and structure

1. Iteration and recursion. Princip *PR1*, na konkrétních algoritmech *ALG*, nikoliv systematicky \emptyset
2. Procedures, functions, methods, and exception handlers Princip *PR1*, *PR2*
3. Concurrency, communication, and synchronization *OSS*, částečně *PSR*

¹<http://www.ets.org/Media/Tests/GRE/pdf/CompSci.pdf>

²<https://cw.felk.cvut.cz/lib/exe/fetch.php/misc/radaoi/publicity/oi-complete.pdf>

- C. Programming languages and notation
 1. Constructs for data organization and program control *PR1*
 2. Scope, binding, and parameter passing *PR1*
 3. Expression evaluation. *Základní principy PR1,ALG?,PAL?,JAG?*
- D. Software engineering
 1. Formal specifications and assertions *SI*
 2. Verification techniques *SI,PSR*
 3. Software development models, patterns, and tools *SI,částečně PII,'patterns' částečně v DS?*
- E. Systems
 1. Compilers, interpreters, and run-time systems \emptyset
 2. Operating systems, including resource management and protection/security *OSS*
 3. Networking, Internet, and distributed systems *OSS,PII*
 4. Databases *DS*
 5. System analysis and development tools *SI,DS,ASS?*

II. COMPUTER ORGANIZATION AND ARCHITECTURE (15%)

- A. Digital logic design
 1. Implementation of combinational and sequential circuits *SPS,NVS?*
 2. Optimization and analysis *SPS,EM?*
- B. Processors and control units
 1. Instruction sets *SPS, částečně NVS,APO*
 2. Computer arithmetic and number representation *SPS,APO, částečně NVS*
 3. Register and ALU organization *APO,SPS*
 4. Data paths and control sequencing *SPS,EM?*
- C. Memories and their hierarchies
 1. Performance, implementation, and management *SPS?,okrajově APO,EM?*
 2. Cache, main and secondary storage *APO*
 3. Virtual memory, paging, and segmentation *APO*
- D. Networking and communications
 1. Interconnect structures (e.g., buses, switches, routers) *APO,částečně OSS*
 2. I/O systems and protocols *částečně OSS, NVS* z pohledu RISC hardware.
 3. Synchronization *OSS,PSR?*
- E. . High-performance architectures

1. Pipelining superscalar and out-of-order execution processors *PAP?*, *není v bakalářské etapě*
2. Parallel and distributed architectures *částečně PII, PAR?*, \emptyset

III. THEORY AND MATHEMATICAL BACKGROUND (40%)

I. Algorithms and complexity

1. Exact and asymptotic analysis of specific algorithms *ALG informuje na příkladech, nikoliv 'analysis', TAL?* \emptyset
2. Algorithmic design techniques (e.g., greedy, dynamic programming, divide and conquer) *ALG, PAL*
3. Upper and lower bounds on the complexity of specific problems *TAL?*
4. Computational complexity, including NP-completeness *TAL?*

II. Automata and language theory

1. Models of computation (finite automata, Turing machines) *JAG, konečné automaty v SPS*
2. Formal languages and grammars (regular and context free) *JAG*
3. Decidability *JAG?*

III. Discrete structures

1. Mathematical logic *LGR*
2. Elementary combinatorics and graph theory *LGR?*
3. Discrete probability, recurrence relations, and number theory *PSI, DMA?*

IV. OTHER TOPICS (5%)

1. numerical analysis *NUM*
2. artificial intelligence *ZUI, BIA*
3. computer graphics *APG*
4. cryptography *TAL?*
5. security \emptyset
6. social issues \emptyset
7. ...

2 Otázky

(podle GRE Computer Science Test Practice Book, 2004³)

Ke každé otázce uvádíme, které předměty by měly absolventům poskytnout znalosti pro její zodpovězení. Kódování je stejné jako v minulé sekci.

Instrukce pro guaranty předmětů: Prosím zkontrolujte, zda váš předmět skutečně umožní na příslušnou otázku odpovědět, případně nám oznamte opak. Pokud narazíte na otázku, kterou váš předmět též pokrývá a není u ní napsán, oznamte nám to též.

Otázka	Předmět
1	<i>PSI?</i>
2	<i>PR1</i>
3	<i>ALG</i> kromě 'tournament sort'
4	\emptyset
5	<i>PR1</i>
6	<i>LGR</i>
7	<i>ALG</i>
8	<i>ALG</i>
9	<i>OSS, DSP?, NVS, PII</i>
10	<i>PAL, DMA, LGR</i>
11	<i>APO?</i>
12	<i>DMA, MA2, TAL</i>
13	<i>DMA?</i>
14	<i>PAL</i>
15	<i>ALG?</i>
16	<i>PII</i>
17	<i>NVS, prSPS</i>
18	<i>KO?</i>
19	<i>LGR</i>
20	<i>APO</i>
21	<i>APO</i>
22	<i>APO</i>
23	<i>LGR</i>
24	<i>PR1, PR2</i>
25	<i>PR1, PR2</i>
26	<i>JAG</i>
27	<i>SPS</i>
28	<i>ALG, DMA, MA2, TAL</i>
29	<i>ALG</i>
30	<i>PR1</i>
31	<i>OSS, PII?</i>

³<http://www.ets.org/Media/Tests/GRE/pdf/CompSci.pdf>

32 částečně *TAL*, \emptyset
33 *APO*
34 *PSI?*
35 *JAG*
36 *SPS*
37 *PAL,KO?*, částečně *LGR,TAL*
38 *PSI?,TAL*
39 *PSI?,TAL*
40 částečně *PSR?*, \emptyset
41 *LGR*
42 *LGR*
43 *LGR*
44 \emptyset
45 *NUM?*
46 *ASS?*
47 *JAG,TAL?*
48 *APR?*
49 \emptyset
50 *LGR,KO?*
51 *LGR,KO?*
52 *DAS?*, částečně *PII*
53 *OSS*, částečně *PII*, okrajově *PSR*,
54 částečně *DS*, \emptyset
55 \emptyset
56 *APO*
57 *PSI*
58 *LGR*
59 *JAP?,JAG*
60 *ALG*
61 *JAG?*
62 *PAL*
63 \emptyset
64 *ALG,DMA?*
65 *DMA*
66 *JAG?*
67 *JPG?,DMA*
68 *TAL*
69 *TAL*
70 *DAG?,DMA*

3 Poznámky

De facto autoři předmětů ALG, NVS a jiných jsou ochotni upravit obsah tak, aby některé další předměty pokrýval.

Obsah předmětu PAL ještě není zapracován.

Z katedry matematiky má reakci jen na pokrytí otázek (druhá část), nikoliv na pokrytí témat (první část).

4 Závěr

Zdá se, že většina témat z GRE CS testu je předměty OI pokryta. Zatím to vypadá, že výjimky jsou:

- bezpečnost (security)
- formální verifikace správnosti programů (např. invarianty)
- možná některé obecné techniky algoritmického návrhu (např. dynamické programování, rozděl a panuj).
- konstrukce překladačů a run-time systémů (např. garbage collection)
- konstrukce a analýza randomizovaných algoritmů
- paralelní a distribuované výpočetní systémy.
- “social issues” (asi není třeba učit)

Je nicméně vidět, že pokrytí témat předměty není zdaleka rovnoměrné. Některé předměty, jako třeba *Algoritmizaci (ALG)* nebo *Logika a grafy (LGR)* pokrývají velkou část znalostí vyžadovaných testem a pravděpodobně nebude možné v nich látku probrat do hloubky kterou GRE CS test předpokládá.