

Arhitectura Sistemelor de Calcul (ASC)
Examinarea finală
Varianta 1
(2022 - 2023)

Anul I, Semestrul I
21 ianuarie 2023
Cristian Rusu

Nume: _____

Prenume: _____

Grupa: _____

Completați aici totul cu majuscule.

Toate răspunsurile sunt în albastru.

Înainte de a începe, citiți cu atenție indicațiile următoare:

- Testul și rezolvarea sa vor fi disponibile online în zilele următoare.
- *Nu aveți voie cu laptop-uri sau alte dispozitive de comunicație.*
- *Nici calculatoarele de buzunar nu sunt permise.*
- *Vă rugăm să vă opriți telefoanele mobile.*
- Pentru întrebările cu răspunsuri multiple/simple folosiți tabelele puse la dispoziție.
- Acest test are 6 enunțuri totalizând 100 de puncte.
- Aveți la dispoziție 120 de minute pentru a completa examinarea.
- Mult succes!

Întrebarea 1. (22 puncte)

Completați tabelul de mai jos cu valorile numerice corecte. Toate numerele sunt naturale pe 12 biți. (Fiecare răspuns corect valorează 1 punct)

binar	octal	zecimal	hexazecimal
0001 1111 1001	771	505	0x1F9
0000 0010 1010	52	42	0x02A
0000 1010 1001	251	169	0x0A9
0001 1011 1100	674	444	0x1BC

a binar	a hexa	b binar	b hexa	a+b zecimal	a+b binar	a+b hexa
0000 0010 1010	0x02A	0001 1001 0100	0x194	446	0 0001 1011 1110	0x1BE

a binar	a hexa	b binar	b hexa	axb zecimal	axb binar	axb hexa
0000 0001 0111	0x017	0000 0001 0000	0x010	368	0 0001 0111 0000	0x0170

Întrebarea 2. (13 puncte)

Completați tabelul de mai jos cu valorile numerice corecte. Toate numerele sunt întregi pe 8 biți. (Fiecare răspuns corect valorează 1 punct)

binar	octal	zecimal	hexazecimal
1100 1110	716	-50	0xCE
1111 1001	771 (sau 371)	-7	0xF9

a zecimal	a binar	a hexa	b zecimal	b binar	b hexa
-26	1110 0110	0xE6	-17	1110 1111	0xEF

produs axb zecimal	produs axb binar	produs axb hexa
442	0000 0001 1011 1010	0x01BA

3.1.	Linus Torvalds
3.3.	Donald Knuth
3.5.	2^{N-1}
3.7.	$2^{N-1} - 1$ strict pozitive și 2^{N-1} strict negative
3.9.	timpul de propagare
3.11.	implementează memorie
3.13.	XYZ
3.15.	(NOT A) NOR (NOT B)
3.17.	virtualizare
3.19.	WAR
3.21.	2.6
3.23.	Dennard
3.25.	0%
3.27.	geometrică
3.29.	CPU time
3.31.	Moore's Law/Simplify/Common Case/Parallel/Pipeline/Predict/Mem Hierarchy/Redundant
3.33.	EXEC

Întrebarea 3. (17 puncte)

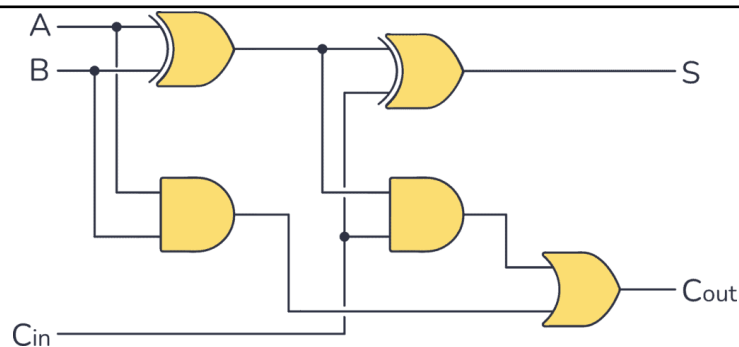
Răspundeți la următoarele întrebări scurte. Completați în tabelul de pe pagina anterioară.

- 3.1. Persoana care a scris primul kernel Linux este ...
- 3.3. Cine a scris seria de volume *The art of computer programming*:
- 3.5. Avem numere naturale pe N biți, câte sunt pare?
- 3.7. Avem numere întregi pe N biți, câte sunt strict negative și câte strict pozitive?
- 3.9. În contextul circuitelor, timpul maxim necesar pentru a produce la ieșirea unui circuit semnale digitale corecte și valide din momentul în care la intrarea unui circuit s-au specificat semnale digitale corecte și valide se numește ...
- 3.11. Care este, în opinia voastră, cel mai mare avantaj al unui circuit secvențial, în comparație cu unul combinațional ...
- 3.13. Simplificați maxim următoarea expresie logică $(XZ + !XY + YZ) * X * (!X + Y)$ (scrieți formula folosind doar AND, OR și NOT) ...
- 3.15. Scrieți următoarea expresie logică $A \text{ AND } B$ folosind doar NOR și NOT ...
- 3.17. Tehnologia care permite rularea mașinilor virtuale este ...
- 3.19. Fie următoarele două instrucțiuni $\%ebx \leftarrow \%ecx * \%eax$, $\%eax \leftarrow \%eax - \%edx$, ce fel de hazard apare în această secvență?
- 3.21. Avem o secvență de cod care este paralelizabilă în proporție de 40%. Presupunând că putem îmbunătăți viteza de calcul pentru secvența paralelă cu $\delta = 5$, care este îmbunătățirea (speed-up) în viteza de execuție?
- 3.23. Legea care ne spune că frecvența procesoarelor este dificil de îmbunătățit semnificativ din cauza puterii electrice necesare și disipării căldurii este ...
- 3.25. În contextul un sistem cu memorie cache, miss rate-ul doar pentru memoria RAM este ...
- 3.27. Pentru a calcula îmbunătățirea medie între raportul timpilor de execuție a unui "program A" și "program B" vom folosi media ...
- 3.29. Timpul (măsurat în ms) pe care un proces îl stă în execuție pe procesor se numește ...
- 3.31. Numiți una dintre cele 8 mari idei din arhitectura calculatoarelor (conform cărții noastre de referință PH book) ...
- 3.33. Syscall-ul pentru a executa un fișier binar ELF sub Linux este:

Întrebarea 4. (14 puncte)

Se dă schema digitală din figura de mai jos. Răspundeți la următoarele întrebări:

- 4.1. (3 puncte) Care sunt funcțiile logice pentru cele două ieșiri?
- 4.2. (3 puncte) Ați văzut la curs formule asemănătoare? Este schema de mai jos echivalentă cu o schemă/circuit de la curs?
- 4.3. (3 puncte) Dacă schema de mai jos este echivalentă cu o schemă de la curs, este schema de mai jos mai eficientă decât cea de la curs?
- 4.4. (5 puncte) Scrieți funcțiile logice pentru ieșiri, și simplificați-le maxim, dacă știm că $B = 1$. Rezultatul final să fie scris folosind doar AND, OR și NOT.



4.1. $S = A \oplus B \oplus C_{in}$ și $C_{out} = AB + (A \oplus B)C_{in}$.

4.2. Este adunarea binară de 1 bit din Cursul 0x04 dar acolo am avut $S = A \oplus B \oplus C_{in}$ și $C_{out} = AB + (A + B)C_{in}$. Expresiile pentru S sunt identice iar cele două expresii pentru C_{out} sunt echivalente (verificare cu tabel de adevăr sau explicând că este din cauza termenului AB).

4.3. Preferăm această variantă cu $C_{out} = AB + (A \oplus B)C_{in}$ pentru că refolosește rezultatul $A \oplus B$ care oricum este obligatoriu necesar pentru S . Varianta din curs calculează $A + B$ în plus și este deci mai ineficientă când vrem ambele valori S și C_{out} (altfel, dacă am vrea să calculăm doar C_{out} am folosi formula din curs, pentru că AND e mai simplu/eficient decât XOR).

4.4. $S = A \oplus 1 \oplus C_{in} = \neg(A \oplus C_{in}) = AC_{in} + \neg A \neg C_{in}$ și $C_{out} = A + (A \oplus 1)C_{in} = A + \neg AC_{in} = A + C_{in}$. Acest calcul este echivalent cu adunarea $a + 11\dots11$ (a plus N biți de 1, dacă a este pe N biți).

Figura este preluată de la <https://www.build-electronic-circuits.com/full-adder/>

Întrebarea 5. (16 puncte)

Vi se dă numărul real 30.5. Realizați următoarele cerințe:

- 5.1. (4 puncte) Să se reprezinte numărul în format IEEE FP pe 32 de biți în hexazecimal.
- 5.2. (4 puncte) Care este următorul număr în format IEEE FP pe 32 de biți după 30.5? Dați reprezentarea hexazecimală și zecimală.
- 5.3. (4 puncte) Care este numărul în format IEEE FP pe 32 de biți înainte de 30.5? Dați doar reprezentarea hexazecimală.
- 5.4. (4 puncte) Care este numărul $\frac{30.5}{2}$ în format IEEE FP pe 32 de biți? Dați reprezentarea hexazecimală și zecimală.

5.1. 0x41F40000.

5.2. 0x41F40001 și $30.5 + 2^{-19}$.

5.3. 0x41F3FFFF.

5.4. 0x41740000 și 15.25.

Întrebarea 6. (18 puncte)

Pentru $n \geq 1$ natural, se dă următoarea formulă recursivă numită Collatz:

$$n \leftarrow \begin{cases} \frac{n}{2}, & \text{dacă } n \text{ este par} \\ 3n + 1, & \text{dacă } n \text{ este impar.} \end{cases} \quad (1)$$

De exemplu, dacă începem de la numărul $n = 12$ și aplicăm repetat formula de mai sus atunci avem secvența: 12, 6, 3, 10, 5, 16, 8, 4, 2, 1. La 1 ne oprim mereu.

Realizați următoarele cerințe:

- 6.1. (2 puncte) Observați că atunci când n este impar vom calcula $3n + 1$ care este mereu par. Astfel, după fiecare pas impar putem realiza automat un pas par. Actualizați formula de mai sus pentru a face calculul în acest fel (echivalent cu doi pași realizați simultan pentru cazul impar). Scopul este să ajungem la 1 cât mai repede.
- 6.2. (4 puncte) Cu noua formulă de actualizare calculați din nou secvența pentru $n = 12$. Noua secvență trebuie să fie mai scurtă decât cea din enunț.
- 6.3. (12 puncte) Scrieți o expresie aritmetică/logică pentru actualizare lui n dar fără “dacă”, adică branchless Collatz. Simplificați maxim expresia. Porniți de la formula găsită la 6.1. Atenție, în toate formulele n rămâne mereu un număr natural, deci rezultatul împărțirii la 2 este mereu un număr natural (rotunjit în jos dacă numărătorul este impar).

6.1. Noua formulă este:

$$n \leftarrow \begin{cases} \frac{n}{2}, & \text{dacă } n \text{ este par} \\ \frac{3n+1}{2}, & \text{dacă } n \text{ este impar.} \end{cases}$$

6.2. Cu noua formulă, pentru $n = 12$ avem: 12, 6, 3, 5, 8, 4, 2, 1.

6.3. Branchless Collatz începe de la noua formulă:

$$\begin{aligned} n &\leftarrow \frac{3n+1}{2}(n \& 1) + \frac{n}{2}!(n \& 1) && \text{[e un MUX cu semnalul de selecție paritatea]} \\ &= \frac{3n}{2}(n \& 1) + \frac{n}{2}!(n \& 1) + \frac{1}{2}(n \& 1) \\ &= \left(n + \frac{n}{2}\right)(n \& 1) + \frac{n}{2}!(n \& 1) + \frac{1}{2}(n \& 1) \\ &= \frac{n}{2} + \frac{2n+1}{2}(n \& 1). \end{aligned}$$

Avem o problemă, când n este impar atunci și $2n + 1$ este impar, deci pierdem 1. Avem corecția:

$$\begin{aligned} n &\leftarrow (n \gg 1) + ((2n + 3) \gg 1)(n \& 1) \\ &= (n \gg 1) + n(n \& 1) + (3 \gg 1)(n \& 1) \\ &= (n \gg 1) + n(n \& 1) + (n \& 1) \\ &= (n \gg 1) + (n + 1)(n \& 1). \end{aligned}$$