University of Nevada, Las Vegas Computer Science 456/656 Spring 2026

CSC 456/656 Spring 2026 Third Examination April 15, 2026

Name:____

The entire test is 355 points.

No books, notes, scratch paper, or calculators. Use pen or pencil, any color. Use the rest of this page and the backs of the pages for scratch paper. If you need more scratch paper, it will be provided.

1. True or False.[5 points each] T = true, F = false, and O = open, meaning that the answer is not known science at this time.

   (i) **T** Every language is a subset of some regular language.

   (ii) **O** If L is any $\mathcal{P}$–TIME language, there is an $\mathcal{NC}$ reduction of the Boolean circuit problem (CVP) to L.

   (iii) **F** Every function that can be mathematically defined is bounded by some recursive function.

   (iv) **F** There are countably many languages over the binary alphabet.

   (v) **T** There are countably many RE languages over the binary alphabet.

   (vi) **T** If L_1 is $\mathcal{NP}$-complete and L_2 is $\mathcal{NP}$ and there is a $\mathcal{P}$–TIME reduction of L_1 to L_2, then L_2 must be $\mathcal{NP}$–COMPLETE.

   (vii) **T** Any computation that can be done by any machine can be done by some Pascal program.

   (viii) **T** Multiplication of integer matrices is $\mathcal{NC}$.

   (ix) **T** There is $\mathcal{P}$–SPACE algorithm which decides SAT.

   (x) **O** Every dynamic program problem can be worked by polynomially many processors in polylogarithmic time.

   (xi) **O** There is a polynomial time reduction of the subset sum problem to 2-SAT.

   (xii) **T** Every $\mathcal{P}$–TIME problem has an $\mathcal{NC}$ reduction to the Circuit Value Problem.

   (xiii) **T** If a language L is accepted by a non-deterministic machine, then L must be accepted by some deterministice machine.

   (xiv) **T** Every context-free language is $\mathcal{NC}$.

   (xv) **T** The language $\{a^n b^n c^n d^n : n \geq 0\}$ is $\mathcal{NC}$.

   (xvi) **T** The context-free grammar equivalence problem is co-$\mathcal{RE}$.

   (xvii) **T** The set of all binary numerals for prime numbers is $\mathcal{P}$–TIME.

   (xviii) **F** The union of any two undecidable languages is undecidable.

(xix) **T** co-$\mathcal{P}$–TIME= $\mathcal{P}$–TIME.

(xx) **T** The complement of any $\mathcal{P}$–SPACE language is $\mathcal{P}$–SPACE.

(xxi) **T** The jigsaw puzzle problem is $\mathcal{NP}$ complete. (Given a set of polygons, can they be fit together to fill a given rectangle.)

(xxii) **T** The complement of any undecidable language is undecidable.

(xxiii) **T** If a Boolean expression is satisfiable, there is a polynomial time proof that it is satisfiable.

(xxiv) **T** If there is a recursive reduction from the halting problem to L, then L must be undecidable.

(xxv) **F** If L is undecidable, there must be a recursive reduction from the halting problem to L.

2. [5 points] Fill in the blank. A language is **decidable** or **recursive** if and only if it is both RE and co-RE.

3. Here is a list of problems or languages. For each problem, enter **T** if it is known to be $\mathcal{NP}$–complete, **F** if it is not known to be $\mathcal{NP}$–complete. [5 points each]

   (a) **T** SAT

   (b) **F** 2-SAT

   (c) **T** 3-SAT

   (d) **F** Rush Hour

   (e) **F** The Boolean circuit problem.

   (f) **F** Integer factoring, using binary numerals.

   (g) **T** The tiling problem. Can you cover a given rectangle with a given set of polygoins.

   (h) **T** The bin packing problem. Given a set of bins, each with a given capacity, and given a set of items, can all the items be packed into the bins?

4. [20 points] Explain how to find the maximum of a list of n integers in logarithmic time using n processors.

   Each of $n/2$ processors finds the maximum of one pair of integers, yielding a list of $O(n/2)$ integers. We repeat that step using $n/4$ processors to obtain a list of $O(n/4)$ integers. After $O(\log n)$ steps, we have just one integer, which is the maximum.

5. [20 points] Give a definition of a recursive real number. (There is more than one correct definition.)

   A real number x is recursive if there is an algorithm which, when run forever, writes the decimal expansion of x.

   Alternatively, a real number x is recursive if the set of all fractions whose values are less than x is a decidable language.

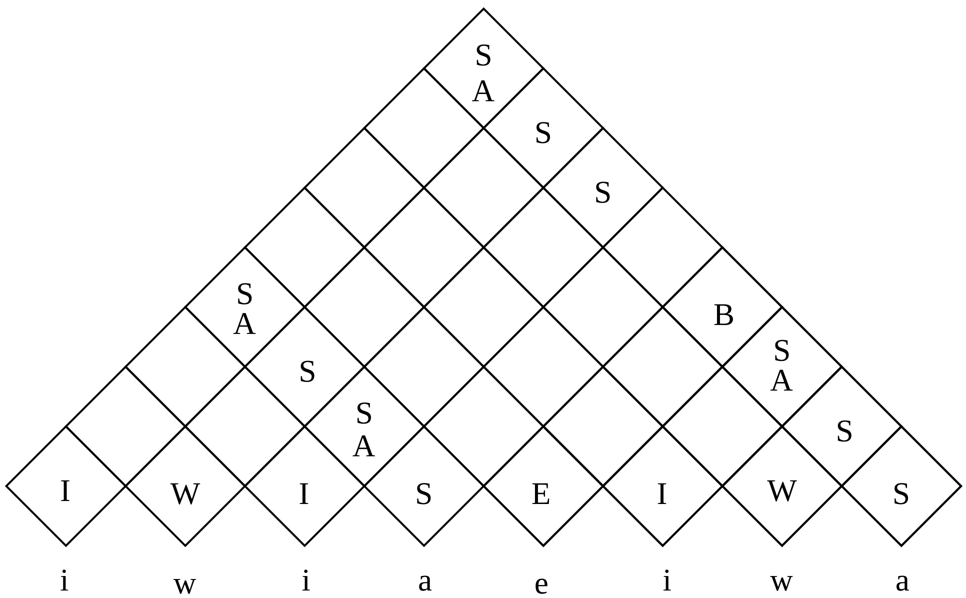
6. [20 points] State the pumping lemma for context-free languages.

For any context-free language L
There exists a positive number p such that
For any $w \in L$ if $|w| \geq p$
There exist strings u, v, x, y, z such that the following four statements hold

1. $w = uvxyz$
2. $|vxy| \leq p$
3. $|v| + |y| > 0$
4. For any $i \geq 0$ $uv^i xy^i z \in L$

7. [20 points] Use the CYK algorithm to prove that *iwiaeiwa* is generated by the following grammar.

1. $S \to a$
2. $S \to WS$
3. $W \to w$
4. $S \to IS$
5. $S \to AB$
6. $A \to IS$
7. $B \to ES$
8. $E \to e$
9. $I \to i$



The top cell contains the start symbol, thus *iwiaeiwa* is generated by the grammar.

8. [20 points] Give the verifier definition of the class $\mathcal{NP}$.

If L is $\mathcal{NP}$, there exists an integer k and a machine M such that, for any $w \in L$ there exists a string v such that M accepts (w, v) in $O(n^k)$ time, and furthermore, if $w \notin L$ and v is any string, M does not accept (w, v).

9. [30 points] Let L be the Dyck language, but where each left parenthesis is written as a and every right parenthesis as b. (This makes grading easier, since if you write parentheses carelessly, they look alike.)

Here is an unambiguous CFG for L.

1. $S \to a_2 S_3 b_4 S_5$
2. $S \to \lambda$

(a) Fill in the action and goto tables for the grammar given above. I have started the tables by writing row 0 and row 4.

	a	b	\$	S
0	$s2$		$r2$	1
1			HALT	
2	$s2$	$r2$		3
3		$s4$		
4	$s2$	$r2$	$r2$	5
5		$r1$	$r1$	

(b) Show the computation of the parser for the input string $aabbab$.

$\$_0$	$aabbab\$$		
$\$_0 a_2$	$abbab\$$		$s2$
$\$_0 a_2 a_2$	$bbab\$$		$s2$
$\$_0 a_2 a_2 S_3$	$bbab\$$	2	$r2$
$\$_0 a_2 a_2 S_3 b_4$	$bab\$$	2	$s4$
$\$_0 a_2 a_2 S_3 b_4 S_5$	$bab\$$	22	$r2$
$\$_0 a_2 S_3$	$bab\$$	221	$r1$
$\$_0 a_2 S_3 b_4$	$ab\$$	221	$s4$
$\$_0 a_2 S_3 b_4 a_2$	$b\$$	221	$s2$
$\$_0 a_2 S_3 b_4 a_2 S_3$	$b\$$	2212	$r2$
$\$_0 a_2 S_3 b_4 a_2 S_3 b_4$	\$	2212	$s4$
$\$_0 a_2 S_3 b_4 a_2 S_3 b_4 S_5$	\$	22122	$r2$
$\$_0 a_2 S_3 b_4 S_5$	\$	221221	$r1$
$\$_0 S_1$	\$	2212211	$r1$

HALT

10. Every language falls into exactly one of these classes.

(a) Known to be $\mathcal{P}$–complete.
(b) Known to be $\mathcal{NC}$.
(c) Known to be $\mathcal{NP}$, but not known to be $\mathcal{P}$–TIME, but not known to be $\mathcal{NP}$–complete.
(d) Known to be $\mathcal{P}$–SPACE, but not known to be $\mathcal{NP}$.
(e) Known to be undecidable and $\mathcal{RE}$ (recursively enumerable).
(f) Known to be decidable, but not known to be $\mathcal{P}$–SPACE.
(g) Known to be undecidable and co-$\mathcal{RE}$.
(h) Not known to be either $\mathcal{RE}$ or co-$\mathcal{RE}$.

In which of the above list of classes is each of these languages or problems?

(i) (b) [5 points] The Dyck language.
(ii) (b) [5 points] The language of binary numerals for multiples of 3, where (b) leading zeros are not allowed.
(iii) (a) [5 points] The Boolean circuit problem.
(iv) (d) [5 points] The set of all winning configurations of the game RUSH HOUR.
(v) (g) [5 points] The context-free grammar equivalence problem.
(vi) (e) [5 points] The halting problem.
(vii) (b) [5 points] Integer matrix multiplication.

11. [20 points] Prove that the halting problem is undecidable.

Proof: By contradiction. Assume the halting problem is decidable. There exists a program H such that $H(M, w)$ is true for any machine (program) M and string w if and only if M halts with input w. Let Q be the following program, which takes any program P has input.

$Q(P)$

If $(H(P, P))$ run forever
Else halt.

Now run Q with input Q. If $H(Q, Q)$, then $Q(Q)$ runs forever, contradiction. If $H(Q, Q)$ is false, then $Q(Q)$ halts, also a contradiction. We conclude that the program H cannot exist, and thus the halting problem is undecidable. ∎