

WAIT FOR INSTRUCTIONS BEFORE BEGINNING

HONOR PLEDGE: "I pledge on my honor that I have not given or received any unauthorized assistance on this examination."

Signature and UID: _____

Print name: _____

- *Write your answers with enough detail about your approach and concepts used, so that the grader will be able to understand it easily.*
- *The sum of the grades is 105, but your grades would be out of 100 (thus you get 5 bonus points by solving all the problems).*
- *Select the best choice for the first 10 problems and mark it by **X** in the table below.*

Problem	1	2	3	4	5	6	7	8	9	10
A										
B										
C										
D										
E										

DO NOT WRITE BELOW THIS LINE

Problems 1-10:		Problem 17:	
Problem 11-15:		Problem 18:	
Problem 16:		Total:	

Multiple-choice Problems (Answer THE BEST CHOICE in the Table of the First Page and NOT HERE):

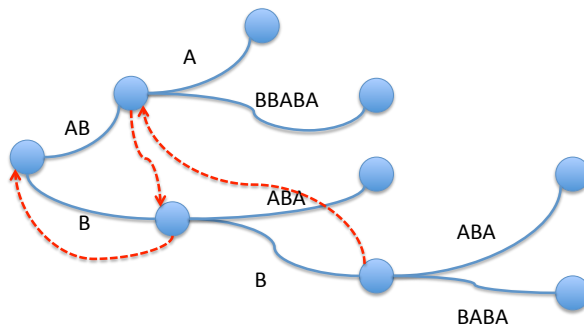
Problem 1. (5 points) Assuming Z-values Z_2, \dots, Z_8 are already computed, how many character comparisons are required to compute Z_9 for string $S = \text{ACCACTACCAG}$ by the linear time Z-algorithm discussed in class?

- a) 0 b) 3 c) 1 d) 9 e) 4

Problem 2. (3 points) In which of the following situations would you use an algorithm like KMP or Boyer-Moore instead of a suffix tree?

- a) There is no situation in which a suffix tree is not preferred
b) Matching many short patterns to a single long target
c) Matching many short patterns to many long distinct targets
d) (b) and (c) e) None of the above

Problem 3. (5 points) Which nodes in the following suffix tree for string $S = \text{ABBBABA}$ have their suffix links incorrectly specified?



- a) AB b) BB c) B
d) (a) and (c) e) None of the above

Problem 4. (2 points) Which of these best represents the relationship between genotype and phenotype?

- a) there is no relationship between genotype and phenotype
b) an individual's phenotype completely determines their genotype
c) an individual's genotype completely determines their phenotype
d) an individual's phenotype partially determines their genotype
e) none of the above

Short Questions (show all derivations as appropriate for full credit):

Problem 11. (6 points) (You can refer to the genetic code figure below). Consider nucleotide sequence S=...CGCATATGAACAAGA...

- How many completely *open reading frames* (ORFs) are there in this sequence (only considering forward strand)
- Write down the aminoacid sequence resulting from translation of one ORF (specify which).
- Specify a *synonymous* nucleotide substitution in this ORF, i.e., does not change aminoacid sequence.
- Specify a *non-synonymous* nucleotide substitution in this ORF.
- Specify a substitution that closes this ORF; write down the resulting aminoacid sequence.

		Second Letter									
		T		C		A		G			
First Letter	T	TTT } TTC } Phe TTA } TTG } Leu	TCT } TCC } TCA } TCG }	TAT } TAC } Tyr TAA } Stop TAG } Stop	TGT } TGC } Cys TGA } Stop TGG } Trp	T	C	A	G	Third Letter	
	C	CTT } CTC } Leu CTA } CTG }	CCT } CCC } CCA } CCG }	CAT } His CAC } CAA } Gln CAG }	CGT } CGC } CGA } CGG }	T	C	A	G		
	A	ATT } ATC } Ile ATA } ATG } Met	ACT } ACC } ACA } ACG }	AAT } Asn AAC } AAA } Lys AAG }	AGT } Ser AGC } AGA } Arg AGG }	T	C	A	G		
	G	GTT } GTC } Val GTA } GTG }	GCT } GCC } GCA } GCG }	GAT } Asp GAC } GAA } Glu GAG }	GGT } GGC } GGA } GGG }	T	C	A	G		

Problem 12. (6 points) Provide a definition of *reproducible data analysis*. Discuss its importance in experimental computational biology. Mention computational tools that can help ensure data analyses are reproducible.

Problem 13. (6 points) Consider pattern $P=AGTCGA$ and target $T=AGCAGTCGAGTC$. Show how the Z-algorithm would be used to find *all* occurrences of P in T in linear time. Calculate all Z-scores required. List those positions in T that need any explicit character comparisons to compute their Z-scores.

Problem 14. (6 points) Consider pattern $P=ATGCATG$ and target $T=ATGCATATGCATGCT$. (1) Calculate all spm_i values for P ; (2) Trace *all* steps used by the KMP algorithm to find all occurrences of pattern P in target T .

Problem 15. (6 points). We showed in class that the worst-case space complexity of suffix *tries* is $O(n^2)$, where n is the size of the string. Discuss the two tricks used to turn a suffix trie to a suffix tree so that only linear space is used in the worst case. Sketch a proof that only linear space is required.

Long Questions (you should always *PROVE THE CORRECTNESS* of your solutions)

Problem 16. (10 points) Prefix-Suffix matching. Give an algorithm that takes two strings α and β , and finds the longest suffix of α that matches a prefix of β . The algorithm should run in time $O(|\alpha| + |\beta|)$.

Problem 17 (15 points) Use a suffix trie to solve the following problem in linear time: Find the longest common substring of strings S and T . Show that your algorithm runs in linear time.

Problem 18. (20 points) Suppose you are given a *dictionary* D of strings s_1, \dots, s_k . Define language L as the set of strings that can be generated by concatenating two or more strings in D . We are interested in determining if D is a *minimal*, that is, no subset of strings s_1, \dots, s_k completely generate language L . You can use keyword trees to address this question

(1). Write an algorithm using a keyword tree to determine if dictionary D is minimal. *Hint: use a keyword tree to determine if a string in D can be generated by concatenating two or more other strings in D .*

(2). Write an algorithm using keyword trees to find a *minimal* subset of D .

(3). **(Extra Credit 5 points)** Answer the two questions above using linear time algorithms. You can assume there exists a linear time algorithm to construct a keyword tree. You are free to extend the data structure as long as your extensions can be constructed along with a failure link.