Note: Write all answer on the exam paper. You can use the blue books to prepare your answer. This quiz is closed book. You are allowed one page of notes.

Problem 0 (worth 1 point): Fill in the 2 blanks in the following sentence.

The two problems that should not be graded on my exam are problem ____ and problem ____.

Part I: (3 points each - 9 points total) Answer 3 out of 4 of the following questions.

1. Answer a, b, c, below True or False: (There is no partial credit for this problem.)

   a. Square matrices have LUP decompositions if and only if they are non-singular.

   b. An $n \times n$ matrix with $n > 2$ which has a 0 in bottom right corner (last row and last column) can never have an LU decomposition.

   c. The anti-diagonal elements of an $n \times n$ are of the form $a_{i,n+1-i}$. (Picture below.) A matrix is anti-diagonal if all elements except the anti-diagonal elements are 0. Let $A$ be an anti-diagonal matrix of rational numbers where all of the anti-diagonal elements of $A$ are not 0.

   T or F: $A$ is invertible and all of its inverses are anti-diagonal.

ANS: a. False - All non-singular matrices have LUP decompositions. The converse is not true.

   b. False - Not true at all - many counterexamples

   c. True - If a matrix has an inverse then it is unique. It is not hard to explicitly write down an inverse of the antidiagonal matrix if the antidiagonal entries are non-zero rationals.

2. Try running the contraction algorithm on each of the three graphs below. For which of the 3 graphs is the probability of finding a min cut of the graph exactly 1/2.

![Graph (a)](image)

![Graph (b)](image)

![Graph (c)](image)
(d) None of the above

\[ C = \frac{3}{4} \times \frac{2}{3} = \frac{1}{2} \]

The others are not 1/2. Specifically for answer a. the graph has probability 4/5 of finding a min cut and for answer b. the probability is 1 of finding a mincut as no matter what order of edges is chosen you end up with a min cut of size 1 after running the algorithm.

3. Assume you are given a long list of 1,000 digits from 0 to 9. You know that one digit, d, appears in the list in more than 3/4 of the places in the list.

Give a probabilistic algorithm to determine the value of d.

Your algorithm should be correct 4/5 of the time, that is the error probability should be at most 1/5. Furthermore, the algorithm should run in constant time, that is some fixed number of steps that is independent of the size of the list.

You should briefly explain why the algorithm you give achieves the desired error and time bound.

---

**ANSWER:**

**Algorithm:**

1. Choose 3 different digits d, e, f uniformly at random from the list.
2. If 2 or 3 of d,e,f are the same output that digit. Of not, output 0.

The time for this algorithm is a fixed number of steps (maybe 7 steps?) and does not depend on how many elements are in the list.

---

Now show the algorithms has the right error bound:

Let \( d_{\text{max}} \) be the digit that appears in 3/4 of the places in the list.

The error probability = the probability that the algorithm gives the wrong answer = the probability that \( d_{\text{max}} \) is not output.

This error probability is bounded by the probability of getting 3 answers not \( d_{\text{max}} \) + the probability of getting 2 of 3 answer which are not \( d_{\text{max}} \).

Now note that the probability of getting 3 answers not \( d_{\text{max}} \leq (1/4)(1/4)(1/4) = 1/64 \).

The probability of getting 2 of 3 answers which are not \( d_{\text{max}} \leq 3((1/4)(1/4)(3/4)) = 27/64 \)

So the error probability is \leq 10/54 and 10/54 < .2.
4. Find an LU decomposition for

\[
A = \begin{pmatrix}
2 & 3 \\
4 & 7
\end{pmatrix}
\]

and also one for

\[
B = \begin{pmatrix}
2 & 4 & 6 \\
0 & 1 & 0 \\
0 & 1 & 0
\end{pmatrix}
\]

ANS: \( A = LU = \begin{pmatrix}
1 & 0 \\
1 & 1
\end{pmatrix} \begin{pmatrix}
2 & 3 \\
0 & 4
\end{pmatrix} \)

and also one for

\[
B = \begin{pmatrix}
2 & 4 & 6 \\
0 & 1 & 0 \\
0 & 1 & 0
\end{pmatrix}
\]

ANS: \( B = LU = \begin{pmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1
\end{pmatrix} \begin{pmatrix}
2 & 4 & 6 \\
0 & 1 & 0 \\
0 & 0 & 0
\end{pmatrix} \)
Part 2: (5 points each - 10 points) Answer 2 out of 3 questions.

5. (i). Give an example non-singular square $3 \times 3$ matrix $A$ which has no LU decomposition with $L$ unit lower triangular and $U$ upper triangular. Explain why $A$ has no LU decomposition.

ANS: Take the matrix

$$A = \begin{pmatrix}
0 & 0 & 1 \\
0 & 2 & 0 \\
3 & 0 & 0
\end{pmatrix}$$

$A$ is non-singular so it has an LUP decomposition.

$A$ has no LU decomposition because if $A = LU$ is such then you can conclude that the first row, first column of $U$ must be a 0 (just write out $L$ and $U$ and see this). And it then follows that the first column of $U$ is all 0's so $U$ is singular and $\det (U) = 0$. But this contradicts that $A$ is nonsingular since $0 = \det (L) \det (U) = \det (A)$.

(ii). Does your example matrix $A$ in part (i) have an LUP decomposition?
If yes, what is its decomposition? If no, then say why not.

For $A$'s decomposition let $P$ be the permutation matrix which flips row 1 with row 3 and gives

$$PA = \begin{pmatrix}
3 & 0 & 0 \\
0 & 2 & 0 \\
0 & 0 & 1
\end{pmatrix}$$

then let $L = \begin{pmatrix} I_n \end{pmatrix}$ and $U = PA$. 

\[ \begin{pmatrix} I_n \end{pmatrix} \]
6. Suppose you find a new algorithm N to multiply an n \times n matrix M by an n \times 1 vector V. Your algorithm N splits M into four n/2 \times n/2 matrices and splits V into two n/2 \times 1 vectors. The picture is: A,B,C,D are n/2 \times n/2, and E and F are n/2 \times 1.

The new method gives a way to do this same multiplication using 3 multiplications of n/2 \times n/2 matrices by n/2 \times 1 vectors plus 2 more multiplications. (We are ignoring any additions here and you may assume n is a power of 2.)

(i). How many multiplications are used by the usual (old) way of multiplying matrices.

ANS: n^2

(ii). We want to compute the complexity of this new algorithm. What is the recurrence relation which arises from this algorithm?

That is: For n \geq 2, define T(n) = the number of integer multiplications carried out in the above algorithm. Write a recurrence equation for T(n). Specifically show how to write T(n) in terms of T(n/2). Don’t forget to include a base case for this recurrence when n = 2.

ANS: T(n) = 3T(n/2)+2 and T(2) = 3+2 = 5.

(iii). What is the value of this recurrence when n = 16?

ANS:

T(2) = 5, T(4) = 3(5)+2 = 17, T(8)=3(17)+2 = 53, T(16)=3(53)+2 = 161.

7. Assume you are given the graph G with 6 vertices as shown below. Also, let S be one specific min-cut in G, also shown below. Define p_S to be the probability of finding the min-cut S in G when you carry out the min-cut contraction algorithm on G.

i. Calculate the value of p_S.

ANS: p_S = (4/6)(3/5)(2/4)(1/3) = 1/15
ii. How many min-cuts does $G$ have?

ANS: 15, all of size 2

iii. If $T$ is a different min-cut of $G$, then does $p_T = p_S$ for all min-cuts $T$ in $G$?

ANS: Yes, all min-cuts $T$ have $p_T = 1/15$

iv. Find a max cut in $G$. How big is it?

ANS: The max cut has size 6 and one such cut is $a,d,e$ and $b,c,f$. 