Here are a few problems of the kind you will be asked to solve on the final exam.

1. You are watching traffic on a busy road and you notice that on average three out of every four trucks on the road are followed by a car, while only one out of every five cars is followed by a truck. What fraction of vehicles on the road are trucks?

Solution: Let the fraction of cars be $x$ and fraction of trucks $y$; then the probability to see a car is $x$ and the probability to see a truck is $y$. These probabilities must satisfy

$$
(x, y) \begin{pmatrix} 4/5 & 1/5 \\ 3/4 & 1/4 \end{pmatrix} = (x, y)
$$

$$
x + y = 1
$$

One of the two homogeneous equations is redundant, so we get a solution by solving

$$
4/5x + 3/4y = x
$$

$$
x + y = 1
$$

Solution: $x = 15/19$ and $y = 4/19$.

2. How many balls in $\mathbb{R}^{1000}$ of radius $1/2$ does it take so that these balls together have the same volume as a unit ball in this space?
We use the fact that $V_{1/2}(1000) = \frac{1}{2}^{1000}V(1000)$.

Thus, $V(1000) = 2^{1000}V_{1/2}(1000)$.

3. (A tiny bit harder) Let $G$ be a strongly connected directed graph, i.e., a graph such that for any two vertices $u, v$ there exists a path from $u$ to $v$. A vertex $w$ in $G$ is periodic if there exists an integer $k$ such that every loop containing $w$ has length divisible by $k$. Show that if one vertex of a strongly connected graph is periodic, then all vertices must also be periodic.

Solution: Assume $v$ is periodic and let $w$ be an arbitrary vertex in $G$. Then there is a path from $v$ to $w$ and a path from $w$ to $v$. These two paths together form a loop containing $v$. Thus, the length of this loop must be divisible by $k$. Consider now an arbitrary loop $l$ containing $w$. Then the path consisting of the path from $v$ to $w$, concatenated with the loop $l$, concatenated with the path from $w$ to $v$ also forms a loop containing $v$ and its length is thus divisible by $k$. However, since the sum of the lengths of the paths from $v$ to $w$ and from $w$ to $v$ is divisible by $k$ also the length of the loop $l$ must itself be divisible by $k$.

4. (Basic probability and the Markov inequality) You are presented with $n$ fair coins, $n \geq 1$; you flip each and if it comes tail you get the coin, if it comes head you do not get it.

(a) Use the Markov inequality to show that the probability to win at least $3/4$ of all coins is not more than $2/3$.

(b) Show that the probability to win at most $1/4$ of all coins is also not more than $2/3$.

(c) Find a formula for finding the exact probability to win at least $k$, $(0 \leq k \leq n)$ out of $n$ available coins and plot its values for all $0 \leq k \leq n$ and $n = 100$. (Your formula can contain summations without a closed form; you can use any software you like for computing and plotting.)

(d) Was Markov inequality useful in this problem?
Solution:

(a) Since the coins are fair, the probability to get a tail at each toss is $p = 1/2$. Thus, the expected number of coins to be won is $E = \sum_{i=1}^{n} (1 \cdot p + 0 \cdot (1 - p)) = n/2$. Thus, if we denote the random variable “number of tails won” by $X$, the Markov inequality for the non-negative random variable $X$ implies

$$P(X \geq 3/4 \cdot n) \leq \frac{E}{3/4 \cdot n} = \frac{n/2}{3/4 \cdot n} = \frac{2}{3}$$

(b) Since the coin is fair, the probability to win $3/4$ of all coins is equal the probability to loose $3/4$ of all coins which happens just in case you win at most $1/4$ of all coins.

(c) The probability to win exactly $k$ coins is $\binom{n}{k} p^k (1 - p)^{n-k}$ if $p = 1/2$. Thus, the probability $P(k)$ to win at least $k$ coins is

$$P(k) = \frac{1}{2^n} \sum_{m=k}^{n} \binom{n}{m}$$

(d) Clearly, the plot of $P(k)$ shows that the probability to win at least $3/4$ of all coins, i.e., 75 coins is negligible, but the Markov inequality here only
shows that such probability is smaller than $2/3$, which is not a very useful estimate.

5. Consider the deterministic Linear Time Algorithm for Order Statistic.

(a) Assume we split the numbers in groups of seven elements. Derive the asymptotic run time of the algorithm, following closely what we did in the case when we split the input numbers into groups of five.

*Hint:* You have to figure out what the recursion should look like and for what $K$ you can derive $T(n) < K \cdot C \cdot n$. In case of splitting into groups of five $K = 11$ worked. If you split into groups of seven you will need a different $K$. Try setting $K$ a variable and see what you get if you substitute this into the right side of the recurrence; it should be easy to figure out what $K$ works.

(b) Assume we split the numbers in groups of three elements. Explain why the proof breaks down.

*Hint:* Show that no $K$ works.

**Solution:**

(a) If we split the numbers into groups of 7, then the middle row has $n/7$ numbers and one half of this row has $n/14$ numbers. Thus, now $4n/14$ many numbers are guaranteed to be smaller or equal to the pivot and as many are larger or equal to the pivot. So after partitioning around the pivot each of the two groups will have at most $10n/14$ many elements. So we are making now one call of the recursive procedure for $n/7$ many elements in the middle row to find a good pivot and another call for at most $10n/14$ many elements, i.e., the recurrence now looks as follows

$$T(n) \leq T(n/7) + T(10n/14) + Cn$$

We will now find a number $K$ such that $T(n) \leq KCn$. Assuming $T(n/7) \leq KCn/7$ and $T(10n/14) \leq KC10n/14$ we now have

$$T(n) \leq T(n/7) + T(10n/14) + Cn \leq KCn/7 + KC10n/14 + Cn = 6KCn/7 + Cn$$

To finish the argument we now need to have $6KCn/7 + Cn \leq KCn$ and for this it is enough to take $K = 7$ because in this case we have

$$T(n) \leq 7Cn/7 + 7C10n/14 + Cn = Cn + 5Cn + Cn = 7Cn$$
(b) Assume now that we group numbers into groups of 3; then the middle row would have \( n/3 \) many elements and its half \( n/6 \) many elements; thus we would have at least \( 2n/6 \) elements smaller than the pivot and \( 2n/6 \) many elements larger than the pivot. So to find the pivot we would call the algorithm on \( n/3 \) many elements and then we would call it on the partition of size at most \( 2n/3 \). Assuming that \( T(n/3) \leq KCn/3 \) and \( T(2n/3) \leq 2KCn/3 \) we would now have

\[
T(n) \leq KCn/3 + 2KCn/3 + Cn = KCn + Cn
\]

To finish the argument we would need that \( KCn + Cn \leq KCn \) which is clearly not true for any \( K \). So we do need to split the numbers in groups of at least 5 (we need an odd number to have a middle row to have simpler counting.)

6. Modify the SkipList data structure so that finding the \( k \)-th smallest element can also be done in expected time \( O(\log n) \).

**Solution:** Assume an element \( a \) has pointers on all levels \( i \) such that \( 0 \leq i \leq k \); for a forward pointer on level \( i \) keep a counter containing the total number of elements between \( a \) and the next element with a pointer on level \( i \). Such a pointer is updated whenever an element is added or deleted.

7. Consider Karger’s MinCut algorithm. How many repetitions of the 4-Contract(\( G \)) would you need to make to guarantee that MinCut is found with a probability of \( 1 - 1/n^2 \) where \( n \) is the number of vertices of \( G \). How does the asymptotic run time of the whole algorithm change?

**Solution:** We have shown that the probability of a success of 4-Contract(\( G \)) is at least \( 1/\log n \); if we repeat it \( K(\log n)^2 \) times the probability \( P \) of a success is at least

\[
P = 1 - \left( 1 - \frac{1}{\log n} \right)^{K(\log n)^2} = 1 - \left( \frac{1}{\log n} \right)^{\log n} = 1 - \frac{1}{n^K}
\]

\[
\approx 1 - e^{-K \log n} = 1 - (e \log n)^{-K} = 1 - n^{-K} = 1 - \frac{1}{n^K}
\]

Thus, to have Karger’s algorithm succeed with a probability of at least \( 1 - 1/n^2 \) we have to run it only \( 2(\log n)^2 \) many times, so asymptotically the algorithm remains an \( O(n^2(\log n)^3) \) algorithm, just the appropriate constant multiplier has doubled in size.
8. The present day “publish or perish” madness in academia involves counting number of papers researchers have published, as well as the number of citations their papers got.

(a) One might argue that not all citations are equally valuable: a citation in a paper that is itself often cited is more valuable than a citation in a paper that no one cites. Design a PageRank style algorithm which would rank papers according to their “importance”, and then use such an algorithm to rank researchers by their “importance”.

(b) Assume now that you do not have information on the citations in each published paper, but instead you have for every researcher a list of other researchers who have cited him and how many times they cited him. Design again a PageRank style algorithm which would rank researchers by their importance.

Solution:

(a) Construct a directed graph with vertices corresponding to individual papers and with edges from \( p_1 \) to \( p_2 \) just in case \( p_1 \) has a citation of \( p_2 \). Now apply the PageRank algorithm (which involves handling dangling vertices and adding weak edges between every two papers etc.). After finding the PageRank of each paper to rank researchers simply add the PageRanks of all of their papers.

(b) Now the vertices of the graph will be the researchers with an edge from \( R_1 \) to \( R_2 \) just in case researcher \( R_1 \) has cited researcher \( R_2 \). This time the edges have weights proportional to the total number of citations of \( R_2 \) by \( R_1 \) (normalised by dividing such a number by the total number of citations \( R_1 \) has made up to this moment.) Apply again the complete PageRank matrix construction and algorithm.

9. You are watching traffic on a busy road and you notice that on average three out of every four trucks on the road are followed by a car, while only one out of every five cars is followed by a truck. What fraction of vehicles on the road are trucks?

Solution: We can assume that the traffic is already(approximately) in a steady state given by the stationary distribution of the corresponding Markov
Process. The states (here they are observations) are “truck” and ”car”; if the previous state was a truck the probability that the next state will be a car is 3/4; thus, the probability that the next state will also be a truck is 1/4. Similarly, if the present state is a car, then the probability that the next state will be a truck is 1/5 and consequently it will be another car with a probability of 4/5. So we are looking for \( c \) (the probability that the current observation is a car) and \( t \) (the probability that the current state is a truck) which satisfy:

\[
\begin{pmatrix}
    c \\
    t
\end{pmatrix} = \begin{pmatrix}
    c \\
    t
\end{pmatrix} \begin{pmatrix}
    4/5 & 1/5 \\
    3/4 & 1/4
\end{pmatrix}; \quad c + t = 1
\]

The solutions of these three linear equations (one of the first two is redundant) are \( c = 15/19 \) and \( t = 4/19 \).

10. You are monitoring Sydney summer weather and you noticed that on average seven out of every 10 pleasant weather days are followed by another pleasant wether day and only one out of every 10 pleasant weather days is followed by a day with a summer shower. You also noticed that on average 2 out of every 15 days with a summer shower are followed by another day with a summer shower and that 4 out of every 9 very hot days are followed by a day with a pleasant wether. You also know that on average out of 90 summer days 50 have pleasant wether, 10 have summer showers and 30 are very hot. Set up the equations which can be used to determine what fraction of days with a shower are followed by a very hot day. Note that you are not asked to solve such a system of equations, just to set up all the equations needed to solve the problem.

**Solution:** This is similar to the car/truck problem. Let \( x \) be the fraction of days with a shower followed by a pleasant day and let \( y \) be the fraction of very hot days followed by a day with a shower. Then we have to have the following equations satisfied:

\[
\begin{pmatrix}
    50/90 \\
    10/90 \\
    30/90
\end{pmatrix} = \begin{pmatrix}
    50/90 \\
    10/90 \\
    30/90
\end{pmatrix} \begin{pmatrix}
    7/10 & 1/10 & 1 - 7/10 - 1/10 \\
    x & 2/15 & 1 - x - 2/15 \\
    4/9 & y & 1 - 4/9 - y
\end{pmatrix}
\]

which is equivalent to

\[
\begin{pmatrix}
    5/9 \\
    1/9 \\
    3/9
\end{pmatrix} = \begin{pmatrix}
    5/9 \\
    1/9 \\
    3/9
\end{pmatrix} \begin{pmatrix}
    7/10 & 1/10 & 2/10 \\
    x & 2/15 & 13/15 - x \\
    4/9 & y & 5/9 - y
\end{pmatrix}
\]

The solutions to above system are \( x = 1/6 \) and \( y = 11/90 \).
11. (a) Recall that a matrix consisting of non-negative reals is *row-stochastic* if in each row all the entries in that row sum up to 1; thus if

\[
M = \begin{pmatrix}
    s_{11} & s_{12} & s_{13} & \cdots & s_{1n} \\
    s_{21} & s_{22} & s_{23} & \cdots & s_{2n} \\
    s_{31} & s_{32} & s_{33} & \cdots & s_{3n} \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    s_{n1} & s_{n2} & s_{n3} & \cdots & s_{nn}
\end{pmatrix}
\]

then for all \(1 \leq i \leq n\) we have \(\sum_{j=1}^{n} s_{ij} = 1\). Assume that \(x^\top = (x_1, x_2, \ldots, x_n)\) is a vector such that \(x_i \geq 0\) and \(\sum_{i=1}^{n} x_i = 1\). Show that then vector \(y = x^\top M\) also has the property that \(y_i \geq 0\) and that \(\sum_{i=1}^{n} y_i = 1\). Thus, if \(x\) is a vector of probabilities of states of a Markov chain, then so is vector \(y\).

(b) In computing the Google PageRank iteratively we start with vector \((1/N, 1/N, \ldots, 1/N)^\top\) giving each page the same initial rank of \(1/N\). Prove that after the iteration has stopped, the sum of the PageRanks of all web pages will be equal to 1.

**Solution:**

(a) Let \(y^\top = x^\top M\) with \(\sum_{i=1}^{n} x_i = 1\). Note that

\[
\sum_{j=1}^{n} y_j = \sum_{j=1}^{n} \sum_{i=1}^{n} x_i M_{i,j} = \sum_{i=1}^{n} \sum_{j=1}^{n} x_i M_{i,j} = \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} x_i = 1.
\]

(b) Apply the previous fact to every step of iteration.

12. Below is given the graph of the Internet one millisecond after the Big Bang.

(a) Construct the corresponding Google matrix with \(\alpha = 7/8\).

(b) Explain what property the PageRank satisfies (i.e., how it is related to the corresponding Google matrix).

(c) Find the PageRank of all nodes (you do not need an iterative algorithm to find the PageRank for such a small matrix; you can solve the corresponding system of linear equations directly, keeping in mind that the
Figure 1: The Internet 1ms after the Big Bang

PageRanks of all pages can be interpreted as probabilities and thus must sum up to 1)

**Solution:** Step one - Matrix $G_0$:

$$
G_0 = \begin{pmatrix}
0 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 & 1/4 \\
1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
0 & 1/3 & 0 & 1/3 & 0 & 0 & 1/3 & 0 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1/3 & 0 & 1/3 & 0 & 0 & 0 & 1/3 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1/3 & 0 & 1/3 & 0 & 1/3 & 0 & 0 & 0
\end{pmatrix}
$$

Step two - fixing the dangling node. There are 8 websites so $N = 8$ and we get
Step 3 - adding the teleportation. Every entry $e_{ij}$ in a row $i$ that was not corresponding to a dangling node is replaced by $\frac{7}{8}e_{ij} + (1 - \frac{7}{8}) \times \frac{1}{8} = \frac{7}{8}e_{ij} + \frac{1}{64}$, so matrix $G$ is

\[
G_1 = \begin{pmatrix}
0 & 1/4 & 1/4 & 1/4 & 0 & 0 & 0 & 1/4 \\
1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
0 & 1/3 & 0 & 1/3 & 0 & 0 & 1/3 & 0 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 \\
1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 \\
1/3 & 0 & 1/3 & 0 & 0 & 0 & 1/3 & 0 \\
1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 \\
1/3 & 0 & 1/3 & 0 & 1/3 & 0 & 0 & 0
\end{pmatrix}
\]

which means that

\[
G = \begin{pmatrix}
1/64 & 7/32 + 1/64 & 7/32 + 1/64 & 7/32 + 1/64 & 1/64 & 1/64 & 1/64 & 7/32 + 1/64 \\
7/32 + 1/64 & 1/64 & 7/32 + 1/64 & 7/32 + 1/64 & 1/64 & 1/64 & 1/64 & 7/32 + 1/64 \\
1/64 & 7/24 + 1/64 & 1/64 & 7/24 + 1/64 & 1/64 & 1/64 & 1/64 & 7/24 + 1/64 \\
1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 \\
7/24 + 1/64 & 1/64 & 7/24 + 1/64 & 1/64 & 1/64 & 1/64 & 1/64 & 7/24 + 1/64 \\
1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 & 1/8 \\
7/24 + 1/64 & 1/64 & 7/24 + 1/64 & 1/64 & 1/64 & 1/64 & 1/64 & 1/64
\end{pmatrix}
\]

Finally, use your favourite linear equations solver to solve

\[
(r(1), r(2), r(3), r(4), r(5), r(6), r(7), r(8)) = (r(1), r(2), r(3), r(4), r(5), r(6), r(7), r(8))G \\
\frac{r(1) + r(2) + r(3) + r(4) + r(5) + r(6) + r(7) + r(8) = 1}
\]

10
for $r(1), \ldots, r(8)$.

You should get something like this:

$r(1) = 0.126617, r(2) = 0.121085, r(3) = 0.154315, r(4) = 0.121085,$
$r(5) = 0.15003, r(6) = 0.101354, r(7) = 0.149437, r(8) = 0.0760767$

Thus, the ordering of webpages according the PageRank is:

$r(3), r(5), r(7), r(1), r(2), r(4), r(6), r(8)$.

13. Prove that if in an irreducible Markov Chain one state is aperiodic, then all states must be aperiodic. Thus, equivalently, assume you have a directed strongly connected graph $G$ which has one vertex $v$ which satisfies that there is NO number $K$ such that the length of every loop containing $v$ is divisible by $K$. Show that in this case all vertices have the same property.

**Hint:** Let $u$ be an aperiodic vertex and $v$ an arbitrary vertex. Since the graph is strongly connected, there is a path $p_1$ from $v$ to $u$ and a path $p_2$ from $u$ back to $v$. Show that for every integer $K$ you can find a path containing $v$ whose length is not divisible by $K$. You might want to make several trips starting at $v$, traversing only $p_1$ and then $p_2$ back to $v$ and just one trip traversing $p_1$ then going through the loop containing $u$ whose length is not divisible by $K$ and then traversing path $p_2$ back to $v$.

**Solution:** Let $u$ be an aperiodic vertex and $v$ an arbitrary vertex. Since the graph is strongly connected, there is a path $p_1$ from $v$ to $u$ and a path $p_2$ from $u$ back to $v$. Assume that $v$ is periodic and so for some integer $K$ every loop containing $v$ has a length divisible by $K$. Consider now a path starting at $v$ that traverses $K - 1$ times the loop consisting of $p_1$ and $p_2$, then traverses again $p_1$ then goes around a loop containing $u$ whose length $l$ is not divisible by $K$ and which exists due to aperiodicity of $u$ and finally traverses $p_2$ back to $v$. The total length of such a lop is

$$(K - 1) \times (\text{length}(p_1) + \text{length}(p_2)) + \text{length}(p_1) + l + \text{length}(p_2) =$$

$$K \times (\text{length}(p_1) + \text{length}(p_2)) + l$$

which is clearly a number not divisible by $K$ because $l$ is not divisible by $K$. 

11
Alternatively, it is also enough to prove that if a Markov chain has one periodic state then all states must be periodic. Assume that \( v \) is periodic with period \( K \) and let \( w \) be any other state (represented as vertices of the associated graph). Consider an arbitrary loop \( \ell \) containing \( w \). There must also be a path \( p_1 \) from \( w \) to \( v \) and a path \( p_2 \) back from \( v \) to \( w \). \( p_1 \) joined with \( p_2 \) forms a loop containing \( v \) thus length(\( p_1 \)) + length(\( p_2 \)) must be divisible by \( K \). Consider also the loop containing \( v \) consisting of the path from \( v \) to \( w \), then loop \( \ell \) and finally the path from \( w \) to \( v \). its total length must also be divisible by \( K \). By subtracting we get that the length of \( \ell \) must also be divisible by \( K \) proving \( v \) is also periodic.

14. One of the most common hash functions used is defined as \( h(x) = x \mod p \) where \( p \) is a prime. This function has a drawback that it is not local, in the sense that two numbers can be close to each other but their hash values can be far away, for example if \( p = 31 \) and \( x_1 = 30 \) and \( x_2 = 31 \) then \( h(x_1) = 30 \mod 31 = 30 \) but \( h(31) = 31 \mod 31 = 0 \). Can you slightly alter this hash function so that numbers which are close to each other have hash values that are also close to each other?

**Solution:** Let \( p \) be a prime; let \( \hat{h}(n) = n \mod 2p \); then we let

\[
h(n) = \begin{cases} 
\hat{h}(n) & \text{if } \hat{h}(n) < p; \\
2p - 1 - \hat{h}(n) & \text{otherwise}
\end{cases}
\]

Note that \( n \mod 2p < 2p \) so the second clause is valid when \( p \leq \hat{h}(n) < 2p \). The idea is that, as \( n \) grows, the usual hash function \( n \mod p \) grows from 0 to \( p - 1 \) and then jumps back to 0 and starts growing again, etc. Instead, the function we defined grows from 0 to \( p - 1 \) and then starts decreasing from \( p - 1 \) back to 0, then starts increasing again, etc. The plot below compares the standard hash function \( n \mod 7 \) with \( h(n) \) as defined above.

15. Assume that you are using a spherical Gaussian of dimension 10,000 with a unit variance in all directions to produce 3 random vectors, \( x, y, z \in \mathbb{R}^{10,000} \). Compute the expected circumference of the triangle whose vertices are given by these 3 vectors. Explain your answer.

**Solution:** As we have seen, the expected length of such a random vector is \( \sqrt{d} \) where \( d \) is the dimension. On the other hand any two such vectors are
mutually orthogonal; thus the size of the vector between two such vertices is, by Pitagoras’ theorem $\sqrt{(\sqrt{d})^2 + (\sqrt{d})^2} = \sqrt{2d}$. Thus, the circumference of the triangle is $3\sqrt{2d}$.

16. Modify the deterministic algorithm for order statistic by splitting the set of input numbers into groups of 9 elements instead of 5 and derive its run time estimate.

Solution: If we split input into 9 groups the middle row will have $\frac{n}{9}$ many elements, and each side of the partition will have at least $\frac{5n}{18}$ many elements. Thus the recurrence is

$$T(n) \leq T\left(\frac{n}{9}\right) + T\left(\frac{13n}{18}\right) + Cn$$

We now prove that $T(n) \leq 6Cn$. Assume that this is true for all $k < n$; then $T\left(\frac{n}{9}\right) \leq \frac{6Cn}{9}$ and also $T\left(\frac{13n}{18}\right) \leq \frac{13\times 6Cn}{18}$; thus

$$T(n) \leq T\left(\frac{n}{9}\right) + T\left(\frac{13n}{18}\right) + Cn$$

$$\leq \frac{6Cn}{9} + \frac{6 \times 13Cn}{18} + Cn$$

$$= \frac{6 \times 15Cn}{18} + Cn$$

$$= 5Cn + Cn$$

$$= 6Cn$$

17. Extend the (doubly linked) SkipList data structure so that finding a median of all elements present in the SkipList runs in constant time.
Solution: Just keep a pointer pointing at the median and a counter for the difference between the number of the elements on the left and the number of elements on the right; update it as items are added or deleted by shifting it one link to the left or to the right if the number of elements at one side exceeds the number of elements at the other side plus 1.

18. Modify the Karger MinCut algorithm so that it produces the correct value of the minimal cut with probability of at least $1 - \frac{1}{n \log n}$ and which runs in time $O(n^2 (\log n)^4)$.

Solution: Just run the 4-contract algorithm $(\log n)^3$ many times, then the probability of a success is equal to

$$1 - \left( 1 - \frac{1}{\log n} \right)^{(\log n)^3} = 1 - \left( \left( 1 - \frac{1}{\log n} \right) \log n \right)^{(\log n)^2}$$

$$\approx 1 - e^{-(\log n)^2} = 1 - \frac{1}{n \log n}$$

19. Consider the following hash function: Let $m$ be a prime; choose $r$ such that the size of the universe $V$ of all possible keys satisfies $|V| < m^{2(r+1)}$. Randomly and independently choose TWO vectors from $\{0, 1, 2, \ldots, m - 1\}^{r+1}$, say they are $\vec{a} = (a_0, a_1, a_2, \ldots, a_r)$ and $\vec{b} = (b_0, b_1, b_2, \ldots, b_n)$. As before, let $\vec{x} = (x_0, x_1, \ldots, x_r)$ be the digits of the representation of $x$ in basis $m$, i.e., such that $x = x_0 + x_1 m + \ldots + x_r m$. Define a hash function $h_{a,b}(x)$ mapping $V$ into $\{1, 2, \ldots, m - 1\} \times \{1, 2, \ldots, m - 1\}$ by

$$h_{a,b}(x) = \left( \sum_{0 \leq i \leq r} x_i a_i \mod m, \sum_{0 \leq i \leq r} x_i b_i \mod m \right) = (\langle \vec{x}, \vec{a} \rangle \mod m, \langle \vec{x}, \vec{b} \rangle \mod m)$$

Show that such a family of functions is universal, i.e., that any two keys collide with probability of $1/m^2$. Note that in this case the size of the hash table is $m \times m$ i.e., with $m^2$ many slots.

Solution: We essentially repeat the proof for a single random vector twice, by showing that for every sequence of integers $a_1, \ldots, a_r$ and every sequence $b_1, \ldots, b_r$ there exist exactly one $a_0$ and one $b_0$ producing a collision.
20. Assume you generate two random vectors \( \vec{a}, \vec{b} \) in a \( d \)-dimensional vector space, where \( d \) is large, by choosing their coordinates independently using a Gaussian of zero mean and a unit variance. Consider vectors \( \vec{d} = \vec{a} - \vec{b} \) and \( \vec{s} = \vec{a} + \vec{b} \). Find the expected values of the lengths \( |\vec{d}| \) and \( |\vec{s}| \) of these two vectors as well as the expected value of the angle between \( \vec{d} \) and \( \vec{s} \).

**Solution:** Note that 
\[
E[\langle \vec{a} - \vec{b}, \vec{a} - \vec{b} \rangle] = E[|\vec{a}|^2] - 2E[\langle \vec{a}, \vec{b} \rangle] + E[|\vec{b}|^2] = E[|\vec{a}|^2] + E[|\vec{b}|^2]
\]
because as we have seen \( E[\langle \vec{a}, \vec{b} \rangle] = 0 \) for any two random independent vectors. Thus, \( E[|\vec{a} - \vec{b}|] = \sqrt{2d} \) because \( E[|\vec{a}|^2] + E[|\vec{b}|^2] = 2d \). Similar calculation shows that also \( E[|\vec{a} + \vec{b}|] = \sqrt{2d} \). Finally, 
\[
E[\langle \vec{a} - \vec{b}, \vec{a} + \vec{b} \rangle] = E[|\vec{a}|^2] - E[\langle \vec{b}, \vec{a} \rangle] + E[\langle \vec{a}, \vec{b} \rangle] - E[|\vec{b}|^2] = d - d = 0.
\]
Thus, \( \vec{a} + \vec{b} \) and \( \vec{a} - \vec{b} \) are mutually orthogonal.

21. Describe the Google matrix produced by taking into account for each webpage and each outgoing link on that webpage the number of times such a link has been clicked on during a (long) period of time. (provided to Google by user’s Chrome browsers). So we no longer assume that every link on a webpage is equally likely to be followed as every other link on the same webpage.

**Solution:** If a webpage \( P_i \) has links to pages \( P_{i1}, \ldots, P_{in} \) only, and if the numbers of clicks on link \( P_i \rightarrow P_{ik} \) is equal to \( c_k \) then \( g_{ijk} = c_k/(c_1 + \ldots + c_n) \) is the entry of the initial matrix \( G_0 \), instead of the usual value 1/n. The rest of the construction of matrices \( G_1 \) and \( G \) is identical to what we described in the class.

22. You are sitting in a veterinary clinic and want to figure out the ratio of dog owners versus cat owners in your neighbourhood, but you do not want to count the large number of dog and cat users visiting the clinic with their pets on that day. Instead you notice that on average every 3 out of 4 dog owners are followed by another dog owner, and only one out of every 3 cat owners are followed by another cat owner. Use such data to estimate the ratio of dog owners to cat owners.

**Solution:** This is just a Markov Chain. We have to find the stationary distribution. Let the probability of seeing a cat be equal to \( c \) and the probability to see a dog at the clinic be \( d \). Then \( c, d \) must satisfy equations
\[(c, d) \begin{pmatrix} 1/3 & 2/3 \\ 1/4 & 3/4 \end{pmatrix} = (c, d) \quad c + d = 1\]

The solution to this system is \( c = 3/11; \ d = 8/11; \) thus the ratio of dog owners to cat owners is \( 8/3. \)