Data Analytics and Machine Learning Group
Department of Informatics
Technical University of Munich

Eexam
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Machine Learning for Graphs and Sequential Data

Exam:	IN2323 / Retake	**Date:**	Wednesday 7th October, 2020
Examiner:	Prof. Dr. Stephan Günnemann	**Time:**	14:15 – 15:30

Working instructions

- This exam consists of **14 pages** with a total of **10 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 38 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
  - all materials that you will use on your own (lecture slides, calculator etc.)
  - **not allowed are any forms of collaboration between examinees and plagiarism**
- You have to sign the code of conduct.
- Make sure that the **QR codes are visible** on every uploaded page. Otherwise, we cannot grade your exam.
- Only write on the provided sheets, **submitting your own additional sheets is not possible**.
- Last two pages can be used as scratch paper.
- All sheets (including scratch paper) have to be submitted to the upload queue. Missing pages will be considered empty.
- **Only use a black or blue color (no red or green)!**
- Write your answers only in the provided solution boxes or the scratch paper.
- **For problems that say "Justify your answer" you only get points if you provide a valid explanation.**
- **For problems that say "Prove" you only get points if you provide a valid mathematical proof.**
- If a problem does not say "Justify your answer" or "Prove" it's sufficient to only provide the correct answer.
- Instructor announcements and clarifications will be posted **on Piazza** with email notifications
- Exam duration - 75 minutes.

Left room from _____ to _____ / Early submission at _____

Problem 1 **Normalizing Flows (5 credits)**

0 1

a) Let $k \in \mathbb{N}$. We consider the transformation $f(\boldsymbol{z}) = (z - 10)^{2k}$ from $\mathbb{R}$ to $\mathbb{R}$. Is this transformation invertible ? Justify your answer. If yes, compute the determinant of its Jacobian and its inverse.

0 1 2

b) We consider the transformation $f(\boldsymbol{z}) = \begin{bmatrix} z_n \\ z_1 \\ : \\ : \\ z_{n-1} \end{bmatrix}$ from $\mathbb{R}^n$ to $\mathbb{R}^n$. Is this transformation invertible ? Justify your answer. If yes, compute the determinant of its Jacobian and its inverse. *Hint: The determinant of one elementary permutation (i.e. a permutation which interchanges any two rows) is* -1

0 1 2

c) We consider the transformation $f(\boldsymbol{z}) = \begin{bmatrix} z_1 + 2z_2 \\ 3z_1 + 4z_2 \\ z_3 - 2z_4 \\ 3z_3 - 4z_4 \end{bmatrix}$ from $\mathbb{R}^4$ to $\mathbb{R}^4$. Is this transformation invertible ? Justify your answer. If yes, compute the determinant of its Jacobian and its inverse.

Problem 2 **Variational Inference (3 credits)**

0
1
2
3

We are performing variational inference in some latent variable model $p_\theta(x, z)$ using the following family of variational distributions $\mathcal{Q}_1 = \{\mathcal{N}(z|\phi, 1) : \phi \in \mathbb{R}\}$.
Assume that

$$p(z) = \mathcal{N}(z|0, 1) \propto \exp\left(-\frac{1}{2}z^2\right)$$
$$p(x|z) = \text{Bernoulli}(x \mid \sigma(z)) \propto \exp\left(xz - \log(1 + \exp(z))\right)$$

and x is known and fixed and $\sigma(z) = \frac{\exp(z)}{1+\exp(z)}$ is the sigmoid function. Does there exist a $q^\star \in \mathcal{Q}_1$, such that $\mathbb{KL}(q^\star(z) \parallel p(z|x)) = 0$? Justify your answer.

Problem 3 Robustness of Machine Learning Models (3 credits)

Consider a trained binary logistic regression model with weight vector $\boldsymbol{w} \in \mathbb{R}^d$ and bias $b \in \mathbb{R}$, where d is the data dimensionality. That is, the predicted probability of a sample $\boldsymbol{x} \in \mathbb{R}^d$ belonging to class 1 is:

$$p(y = 1|\boldsymbol{x}) = \sigma(\boldsymbol{w}^T\boldsymbol{x} + b),$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the logistic sigmoid function. An input sample is assigned to class 1 if $p(y = 1|\boldsymbol{x}) > 0.5$, else it is assigned to class 0.
We would like to perform **robustness certification** of the logistic regression model using its **Lipschitz constant**. That is, we want to certify that the predicted class of the input sample does not change w.r.t. some radius in the input space.

0 1

a) Briefly explain why it is sufficient to consider $F(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$, i.e. the model without the sigmoid activation function $\sigma(\cdot)$ for this purpose.

0 1 2

b) Derive the (smallest possible) Lipschitz constant of $F(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + b$ w.r.t. the L_2 norm. Justify your answer.

only need to know which side the decision boundary is mapped

$$\|F(x_1) - F(x_2)\|_2 \leq K \|x_1 - x_2\|_2$$

$$\|w^T x_1 - w^T x_2\|_2 \leq K \|x_1 - x_2\|_2$$

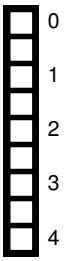
$$a = x_1 - x_2$$

$$K \geq \frac{\|w^T a\|_2}{\|a\|_2}$$

$$K_{max} = \|w^T\|$$

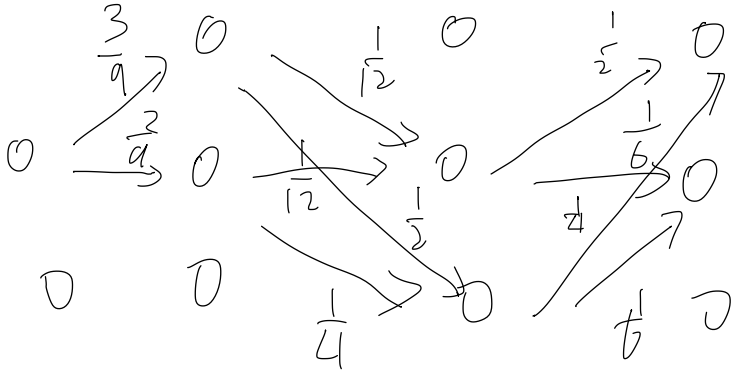
Problem 4 **Hidden Markov Model (4 credits)**

Consider an HMM where hidden variables are in $\{1, 2, 3\}$ and observed variables are in $\{a, b\}$. Let the model parameters be as follows:



$$\pi = \begin{bmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{bmatrix}, \qquad A = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{bmatrix}, \qquad B = \begin{bmatrix} 1 & 0 \\ \frac{2}{3} & \frac{1}{3} \\ 0 & 1 \end{bmatrix}$$

Assume that the sequence $X_{1:3} = [aba]$ is observed. Find the most probable sequence $[Z_1, Z_2, Z_3]$ and compute its likelihood. Justify your answer.


3/9
3/9
1/12
1/12
1/2
1/4
1/2
1/6
1/4
1/4

Problem 5 **Temporal Point Process (3 credits)**

0 1 2 3

Consider a temporal point process defined on the interval $[0, 10\pi]$ with the conditional intensity function

$$\lambda^*(t) = \sin(t) + 2$$

What is the expected number of events that will be generated from this TPP on the interval $[0, 2\pi]$? Justify your answer.

$$\text{井} = \int_0^{2\pi} x^{*}(t)\,dt$$

$$= -\cos t + 2t \Big|_0^{2\pi}$$

$$-1 + 4\pi - [-1] = 4\pi$$

Problem 6 **Neural Network approaches for temporal data (4 credits)**

0
1
2
3
4

We trained the following models to produce latent representations:

- M1: Skip-gram word2vec model
- M2: LSTM
- M3: Self-attention without positional encoding
- M4: Self-attention with positional encoding
- M5: A convolutional NN which produces the latent representation at time t based on the input at time t and $t-1$.

At **inference time**, we use the following 6 sentences as inputs:

- S1: "I left"
- S2: "They left"
- S3: "I left yesterday"
- S4: "I go left"
- S5: "left I go"
- S6: "go left"

For each model (i.e. M1, M2, M3, M4, M5), which sets of sentences (e.g. (S1, S2), (S3, S4, S6)) are guaranteed to produce the same latent representation for the word "left" ? Note that the answer may be zero, one or more sets of sentences. Justify your answer.

$M_1 \Rightarrow (s_4, s_6)$ ✗ embedding are same region of training

$M_2 \Rightarrow (s_1, s_3)$ ✓

$M_3 \Rightarrow (s_4\ s_5\ s_6)$

$M_4 \Rightarrow ($ ✓

$M_5 \Rightarrow (s_4, s_6)\ (s_1, s_3)$ ✓ ✓

Problem 7 PageRank Lollipop (7 credits)

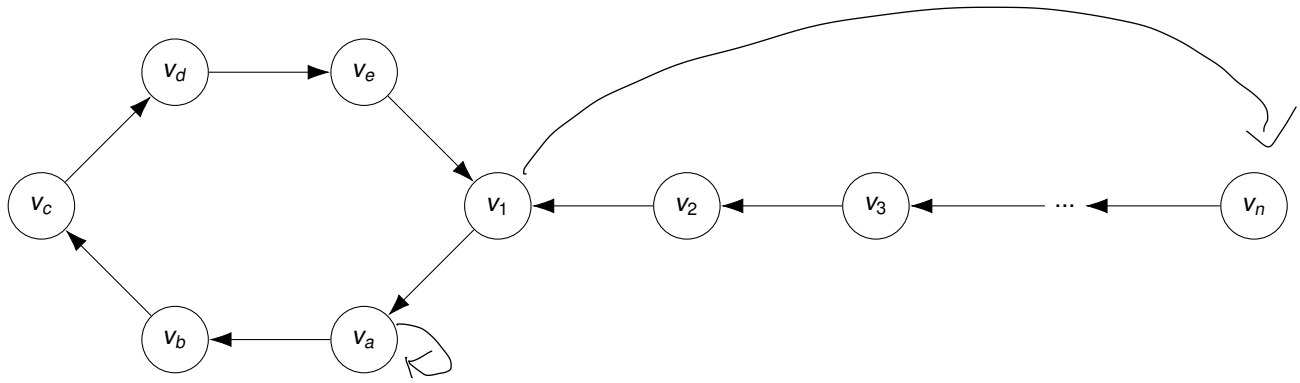


Figure 7.1: A directed "lollipop" graph with a tail of length n

Consider the directed, unweighted graph in Figure 7.1 with a "head" consisting of the six nodes v_1, and v_a through v_e. Its "tail" consists of n nodes $v_1, \ldots, v_n$ where n is a parameter. Note, that we consider v_1 to be part of the head as well as the tail.

0
1
2

a) Which of the PageRank problems of *dead end*, *spider trap*, and *periodic states* apply here? Justify your answer. For each problem that applies, give a set of edges (at most 3 per problem) that would resolve that problem if they were inserted.

~~Dead end~~ ✓ spide trap.

from $\{v_{n-1} \dots v_1\}$ can't go back v_n.

go into (via v_e) can't go out

Open... stay

go back stay

b) In the following, we want to compute the topic-sensitive PageRank of the nodes. For that, we introduce the topic-sensitive teleport vector π where the topic is the "tail", i.e.

$$\pi_a = ... = \pi_e = 0 \quad \text{and} \quad \pi_1 = ... = \pi_n = \frac{1}{n}.$$

State each node's PageRank equation for a teleport probability of $1 - \beta$.

0 1 2

c) Derive the sum of the PageRank of the "head" of the graph $v_1, v_a, ..., v_e$ as a function of n. Justify your answer.

Reminder: $\sum_{i=0}^{k} \beta^i = \frac{1 - \beta^{k+1}}{1 - \beta}$ for $\beta \neq 1$

0 1 2 3

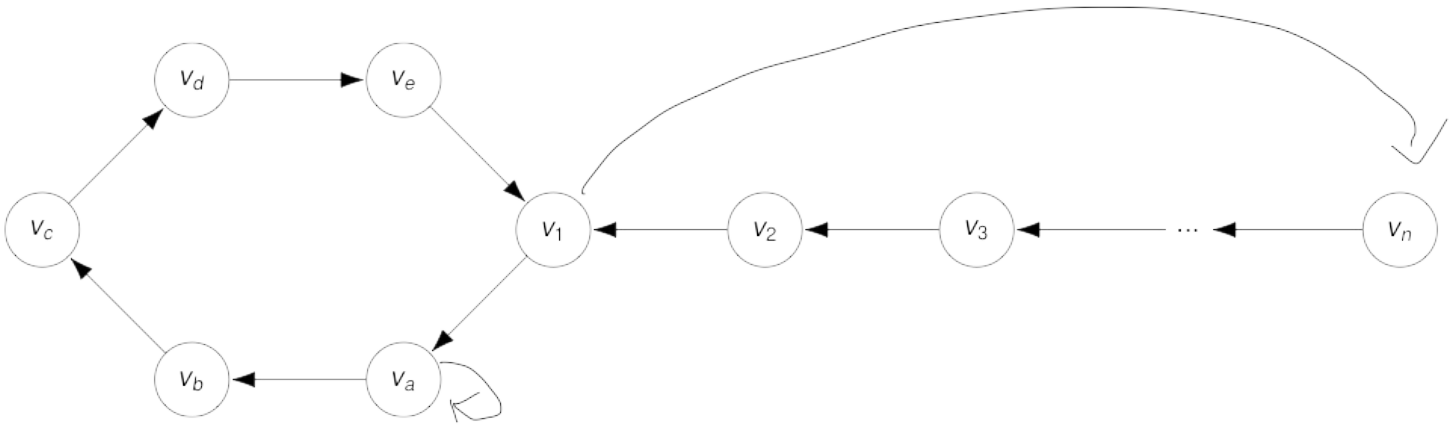


Figure 7.1: A directed "lollipop" graph with a tail of length n

$$r_n = (1-\beta)\cdot\frac{1}{n}$$

$$r_i = \beta\, r_{i+1} + (1-\beta)\cdot\frac{1}{n} \qquad i \in \{2, \dots, n-1\}$$

$$r_1 = \beta\,(r_e + r_2)$$

$r_j =$ ~~$\beta($~~ . $r_a \quad r_b \quad r_c$ - -

$r_a = \beta r_1 \quad r_b = \beta r_a \quad r_c = r r_b \quad r_d = \beta r_c \quad r_e = \beta r_d \quad r_1 = \beta(r_e + r_2) + \frac{1-\beta}{n}$

$r_b = \beta^2 r_1$

$r_c = \beta^3 r_1$

$r_d = \beta^4 r_1$

$r_e = \beta^5 r_1$

$$\Rightarrow r_1 = \beta\left(\beta^5 r_1 + r_2\right) + \frac{1-\beta}{n}$$

$$(1-\beta^6)\, r_1 = \beta r_2 + \frac{(1-\beta)}{n}$$

$$r_1 = \frac{\beta r_2 + \frac{(1-\beta)}{n}}{1-\beta^6}$$

$$r_1 + \dots r_e = \sum_{i=0}^{5} \beta^i r_1$$

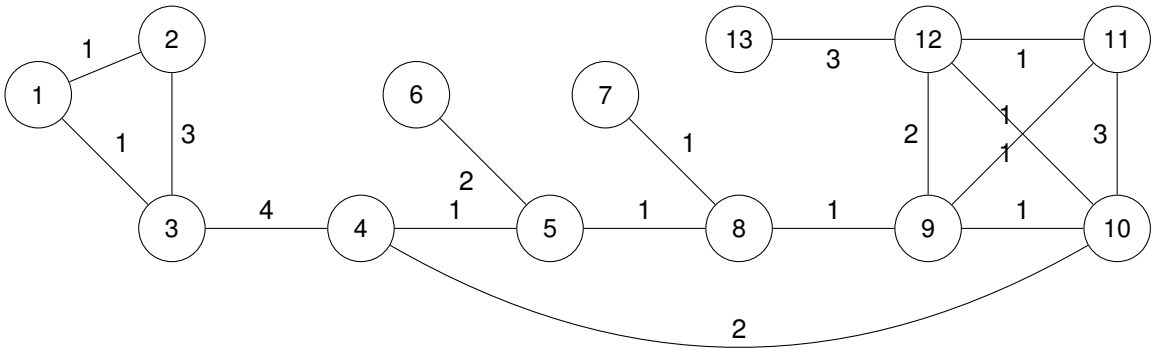
$$= r_1 \cdot \frac{1-\beta^6}{1-\beta}$$

$$= \frac{\beta r_2 + \frac{(1-\beta)}{n}}{1-\beta}$$

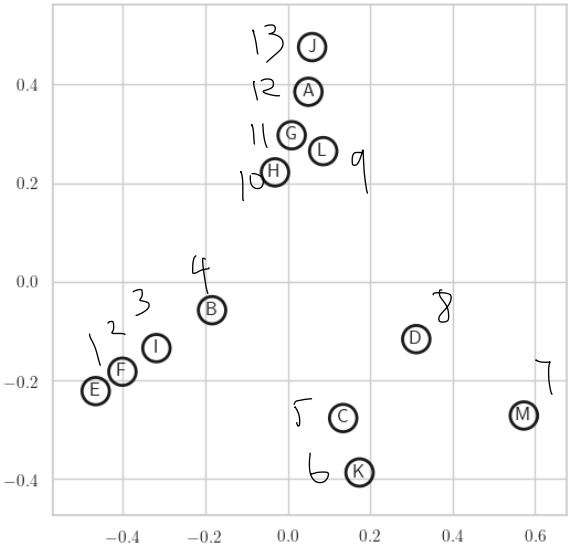
Problem 8 **Spectral Embeddings (3 credits)**

0
1
2
3

We use spectral embedding to map the following undirected, weighted graph into 2-dimensional space.



The following plot shows the embeddings as given by the eigenvectors belonging to the second and third smallest eigenvalues of the unnormalized Lapalacian.



Fill out the following table that maps nodes in the graph to points in the plot.

Node	1	2	3	4	5	6	7	8	9	10	11	12	13
Point													

Problem 9 Graph Neural Networks (4 credits)

a) Consider the following graph neural network.

0
1
2

$$\boldsymbol{Z} = \sigma\left(\sigma\left(\hat{\boldsymbol{A}}\sigma\left(\hat{\boldsymbol{A}}\sigma\left(\hat{\boldsymbol{A}}\boldsymbol{X}\right)\boldsymbol{W}_a\right)\right)\boldsymbol{W}_b\right)\boldsymbol{W}_c$$

$\boldsymbol{X} \in \mathbb{R}^{N\times D}$ are the node features, $\boldsymbol{Z}$ are the node predictions, σ is the sigmoid function, $\boldsymbol{W}_x$ are weight matrices of appropriate dimensions and $\hat{\boldsymbol{A}} = \tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}$ is the propagation matrix as defined for GCNs, i.e. $\tilde{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{I}$ is the adjacency matrix with added self-loops and $\tilde{\boldsymbol{D}}_{ii} = \sum_{j=1}^{N}\tilde{\boldsymbol{A}}_{ij}$ is the degree matrix of the amended adjacency matrix.
Give the transformation and propagation depths of this model. Justify your answer.
Hint: We define

- *propagation depth* as the maximum distance that a model propagates information in the graph,
- *transformation depth* as the number of non-linear transformations that are applied to the features.

b) Write down a formula for the node predictions $\boldsymbol{Z}$ of a graph neural network with propagation depth 2 and transformation depth 2 where the transformations use fully connected layers with weight matrices W_i and biases b_i.
Note: You can assume that operations such as matrix-vector summation broadcast as expected.

0
1
2

true error depth = 4
4 sigmoid function

propagation depth = 3
only 3 $\hat{A}$

$$Z = \hat{A}\hat{A}G\left(G\left(XW_1 + b_1\right)W_2 + b_2\right)$$

Problem 10 Randomized Smoothing on Graph Neural Networks (2 credits)

Consider an arbitrary but fixed graph neural network $f(\boldsymbol{X})$ as base classifier. Here, $\boldsymbol{X}$ *denotes the adjacency matrix* (i.e. the features are assumed to be constant).
For the the smooth classifier $g(\boldsymbol{X})$ we use the randomization scheme $\phi(\mathbf{x})$:

$$g(\mathbf{x})_c = \mathcal{P}(f(\phi(\mathbf{x})) = c) = \sum_{\tilde{\mathbf{x}} \text{ s.t. } f(\tilde{\mathbf{x}})=c} \prod_{i=1}^{n^2} \mathcal{P}(\tilde{\mathbf{x}}_i|\mathbf{x}_i) \tag{1}$$

with

$$\mathcal{P}(\tilde{\mathbf{x}}_i|\mathbf{x}_i) = \begin{cases} p & \tilde{\mathbf{x}}_i = 1 - \mathbf{x}_i \\ 1-p & \tilde{\mathbf{x}}_i = \mathbf{x}_i \end{cases} \tag{2}$$

Note that in contrast to the lecture here we do not distinguish between a probability for deleting p_d or adding p_a an edge. We simply use the "flip" probability p.

Furthermore, we consider the special case of $p = 0.5$.

0 ☐
1 ☐

a) What is the probability $\mathcal{P}(\tilde{\mathbf{x}}|\mathbf{x})$ for a directed graph or is there not enough information to determine it? Justify your answer.

0 ☐
1 ☐

b) We are given the prediction for the original input $f(\boldsymbol{X}) = c$ and for a slightly different version $f(\boldsymbol{X}') \neq c$.
What does the randomization scheme entail about the (exact) prediction of the smooth classifier $g(\boldsymbol{X})_c$ and $g(\boldsymbol{X}')_c$ (i.e. no Monte Carlo approximation)? Specifically, how do the probabilities for class c of the smooth classifier $g(\boldsymbol{X})_c$ and $g(\boldsymbol{X}')_c$ relate?

1. $g(\boldsymbol{X})_c = g(\boldsymbol{X}')_c$
2. $g(\boldsymbol{X})_c < g(\boldsymbol{X}')_c$
3. $g(\boldsymbol{X})_c > g(\boldsymbol{X}')_c$
4. There is not enough information to determine how $g(\boldsymbol{X})_c$ and $g(\boldsymbol{X}')_c$ relate

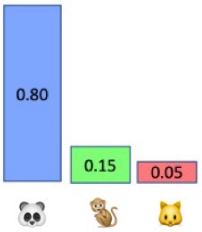
Justify your answer.

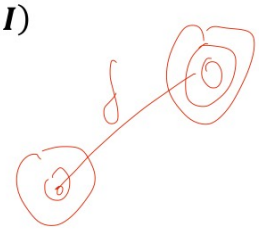
$$p(\tilde{x}_i \mid x_i) = \prod_{i=1}^{n^2} p(\tilde{x}_i^2 \mid x_i)$$

$$\simeq \left(\frac{1}{2}\right)^{n^2}$$

The output of the smoothed classifier $g(\mathbf{x})$ is a vector with entries

$$g(\mathbf{x})_c = \mathbb{P}_{\boldsymbol{\epsilon}}(f(\mathbf{x}+\boldsymbol{\epsilon}) = c) \text{ where } \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$$





Now denote with $c^* = \arg\max_c g(\mathbf{x})_c$ the most likely class and with $p_{\mathbf{x}}^* = g(\mathbf{x})_{c^*}$ the probability of observing c^* (In the example c^*= 🐼 and $p_{\mathbf{x}}^*$ = 0.8)

Goal: We want to certify that for any admissible perturbation $\boldsymbol{\delta}$ it holds

$$\arg\max_c g(\mathbf{x}+\boldsymbol{\delta})_c = c^* \quad \text{for all } \|\boldsymbol{\delta}\|_2 \leq r$$

$g(x)_c$ will only return the probability of class c its independ on input x

Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.