

به نام خدا



دانشگاه تهران  
پردیس دانشکده‌های فنی  
دانشکده برق و کامپیوتر



درس هوش مصنوعی قابل اعتماد

مدرس: دکتر مصطفی توسلی‌پور

تمرین شماره ۳

آذر ماه ۱۴۰۴

۳.....	سوال اول: Energy-Based Models
۴.....	بخش اول: سوالات تئوری
۴.....	زیربخش اول (۴ نمره)
۴.....	زیربخش دوم (۵ نمره امتیازی)
۴.....	زیربخش سوم (۶ نمره)
۵.....	بخش دوم: سوالات پیاده سازی
۵.....	زیربخش اول: بارگذاری دیتاست و آماده سازی داده (۵ نمره)
۶.....	زیربخش دوم: پیاده سازی معماری و آموزش مدل (۱۵ نمره)
۹.....	زیربخش سوم: تولید تصویر (۱۰ نمره)
۹.....	زیربخش چهارم: نوینزدایی تصویر نویزی (۱۵ نمره)
۱۱.....	سوال دوم: Score-Based Models
۱۲.....	بخش اول: سوالات تئوری
۱۲.....	زیربخش اول (۵ نمره)
۱۲.....	زیربخش دوم (۶ نمره)
۱۲.....	زیربخش سوم (۴ نمره)
۱۳.....	زیربخش چهارم (۵ نمره)
۱۳.....	بخش دوم: سوالات پیاده سازی
۱۳.....	زیربخش اول: مدل پایه (۲۰ نمره)
۱۵.....	زیربخش دوم: مدل شرطی (۱۵ نمره)
۱۷.....	نکات تحویل

## سوال اول: ENERGY-BASED MODELS

مدل‌های مبتنی بر انرژی<sup>۱</sup> رویکردی انعطاف‌پذیر در یادگیری ماشین هستند که به جای تعریف صریح یک توزیع احتمال، یک تابع انرژی روی فضای ورودی تعریف می‌کنند. این تابع انرژی به عنوان ابزاری برای نشان دادن «سازگاری» ورودی با ساختار آموخته‌شده مدل عمل می‌کند، بدون آن که نیاز به نرمال‌سازی یا محاسبه احتمال دقیق داشته باشد. چنین مدلی تنها یک منظره انرژی می‌سازد که شکل آن توسط داده‌های آموزشی تعیین می‌شود، و همین منظره پایه‌ی بسیاری از عملیات بعدی قرار می‌گیرد: از نمونه‌سازی و بازسازی گرفته تا نويززادایی و تحلیل رفتار داده‌ها در فضاهای پیچیده. این نگاه، EBMها را به ابزاری مناسب برای مدل‌سازی انعطاف‌پذیر و مفهومی تبدیل کرده است، بی‌آن که مقید به فرم‌های بسته‌ی توزیع‌های احتمالاتی باشند.

---

<sup>۱</sup> Energy-Based Models (EBMs)

## بخش اول: سوالات تئوری

### زیربخش اول (۴ نمره)

اگر بخواهیم یک مدل مبتنی بر انرژی استفاده کنیم که صورت یک مرد جوان را تولید کنیم، برای آموزش راحت تر مدل مبتنی بر انرژی از چه روشی می توانیم استفاده کنیم؟

### زیربخش دوم (۵ نمره امتیازی)

ابتدا روش rejection sampling را توضیح دهید سپس بیان بدارید:

اگر بخواهیم از توزیع  $P(x)$  که توزیع یک مدل مبتنی بر انرژی است و بر روی مجموعه داده MNIST آموزش داده شده است، به کمک روش rejection sampling نمونه گیری کنیم. مقدار optimal acceptance rate چقدر خواهد بود؟

(می دانیم  $\tilde{P} = \mathcal{N}(\mu, \sigma_p^2 I)$  که  $P(x) = \frac{\tilde{P}(x)}{Z}$  و  $Q = \mathcal{N}(\mu, \sigma_q^2 I)$  و همچنین  $\sigma_q = 1.01\sigma_p$ )

### زیربخش سوم (۶ نمره)

مدل های مبتنی بر انرژی خانواده ای انعطاف پذیر از مدل های مولد هستند که توزیع احتمال را به کمک یک تابع انرژی  $f_\theta(x)$  به صورت زیر تعریف می کنند:

$$p_\theta(x) = \frac{e^{f_\theta(x)}}{Z(\theta)}$$

که در آن  $Z(\theta) = \int e^{f_\theta(x)} dx$  تابع پارتیشن است.

برای آموزش این مدل ها به روش بیشینه درست نمایی<sup>۲</sup>، نیاز به محاسبه گرادیان تابع Log-Likelihood داریم. اثبات می شود که این گرادیان برابر است با :

---

Maximum Likelihood <sup>۲</sup>

$$\nabla_{\theta} \log p_{\theta}(x_{train}) = \nabla_{\theta} f_{\theta}(x_{train}) - E_{x_{sample} \sim p_{\theta}} [\nabla_{\theta} f_{\theta}(x_{sample})]$$

الف) با توجه به رابطه بالا، فرآیند آموزش سعی دارد چه تغییری در مقدار  $f_{\theta}$  (انرژی) برای داده‌های آموزشی ( $x_{train}$ ) و نمونه‌های تولیدی مدل ( $x_{sample}$ ) ایجاد کند؟

ب) محاسبه ترم دوم (امید ریاضی روی  $x_{sample}$ ) در عمل با چه چالش محاسباتی روبرو است؟ روش  $CD^3$  چگونه سعی می‌کند این چالش را برطرف کند؟

## بخش دوم: سوالات پیاده سازی

در این بخش قصد داریم یک مدل مبتنی بر انرژی را بر روی دیتاست MNIST آموزش دهیم تا بعد از آن بتوانیم با انجام تعدادی Inference از این مدل‌ها، بیشتر با کاربرد و مفهوم مدل‌های مبتنی بر انرژی آشنا شویم و در این بین نگاهی هم به کاربرد و استفاده نمونه‌گیری با کمک الگوریتم Langevin می‌اندازیم.

### زیربخش اول: بارگذاری دیتاست و آماده‌سازی داده (۵ نمره)

- دیتاست MNIST را با استفاده از torchvision.datasets.MNIST بارگذاری کنید (train و test جداگانه).
- تبدیل‌های لازم را اعمال کنید تا هر تصویر به شکل tensor با ابعاد  $1 \times 28 \times 28$  در بازه  $[0,1]$  باشد.
- برای داده‌های train و test، DataLoader تعریف کنید. (مثلاً batch size حدود ۶۴)
- یک تابع ساده برای نمایش تعدادی تصویر از یک batch بنویسید (مثلاً  $2 \times 8$  تصویر).
  - تعدادی تصویر تصادفی از داده‌های train و test انتخاب کرده و نمایش دهید.
  - خروجی مورد انتظار:
- یک سلول که چند تصویر تصادفی از داده‌های train را نمایش می‌دهد.
- یک سلول که چند تصویر تصادفی از داده‌های test را نمایش می‌دهد.

<sup>۳</sup> Contrastive Divergence

## زیربخش دوم: پیاده سازی معماری و آموزش مدل (۱۵ نمره)

می‌خواهیم یک مدل کانولوشنی تعریف کنیم که با گرفتن عکس، در خروجی یک عدد به عنوان انرژی به آن اختصاص دهد.

- با استفاده از جدول ۱ مدل را ایجاد کنید تا در مراحل بعدی آن را آموزش بدهیم.
  - تابع Langevin sampling را بنویسید. (پیشنهاد: تابع را طوری بنویسید که در صورت نیاز بتوانید مقدار  $x$  شروع را نیز به آن بدهید).
- در الگوریتم ۱ سودوکد آموزش یک گام مدل بیان شده است.
- مدل را براساس سودوکد داده شده در الگوریتم ۱ آموزش دهید. (مقادیر پیشنهادی،  $\text{epoch} = 10$  و  $\lambda = 1e-3$ ) (در هنگام آموزش مدل مقادیر شروع در Langevin sampling، نویزهای uniform بین  $[0,1]$  هستند).
- هنگام آموزش مدل، مقادیر loss، میانگین انرژی real ( $E_{\text{real}}$ ) و میانگین انرژی fake ( $E_{\text{fake}}$ ) را در هر چند صد step یا هر epoch چاپ کنید.
- در اوایل آموزش، وسطای آموزش و آخر آموزش هر بار تعدادی تصویر از تصاویر ایجاد شده با Langevin sampling در طول آموزش را چاپ کنید. (حداقل یک مرتبه در ابتدای آموزش، یک مرتبه در وسط آموزش و یک مرتبه در انتهای آموزش کافی است).
- بعد از اینکه مدل آموزش دید با استفاده از Langevin sampling تعدادی از تصاویر train را بردارید و چاپ کنید. سپس هر کدام از همان تصاویری که برداشتید را به عنوان نقطه شروع به Langevin sampling بدهید و تصاویر تولید شده را نیز چاپ کنید.

### جدول ۱. مشخصات مدل EBM

Row	Configuration	Input Shape	Output Shape
0	—	(B, 1, 28, 28)	(B, 1, 28, 28)
1	Conv2d(1→32, kernel=3, stride=1, padding=1) + LeakyReLU(0.2)	(B, 1, 28, 28)	(B, 32, 28, 28)
2	Conv2d(32→32, kernel=4, stride=2, padding=1) + LeakyReLU(0.2)	(B, 32, 28, 28)	(B, 32, 14, 14)
3	Conv2d(32→64, kernel=3, stride=1, padding=1) + LeakyReLU(0.2)	(B, 32, 14, 14)	(B, 64, 14, 14)
4	Conv2d(64→64, kernel=4, stride=2, padding=1) + LeakyReLU(0.2)	(B, 64, 14, 14)	(B, 64, 7, 7)
5	Conv2d(64→128, kernel=3, stride=1, padding=1) + LeakyReLU(0.2)	(B, 64, 7, 7)	(B, 128, 7, 7)
6	AdaptiveAvgPool2d(output_size=1)	(B, 128, 7, 7)	(B, 128, 1, 1)
7	reshape to vector (Flatten)	(B, 128, 1, 1)	(B, 128)
8	Linear(128→1) (FullyConnected)	(B, 128)	(B,)

### الگوریتم ۱. سودوکد آموزش یک گام مدل

```

B := number of images in x_real (Batch size)

# 1. Compute energy of real data
E_real := E_theta(x_real)          # vector of size B

# 2. Generate fake samples using Langevin dynamics
x_fake := LANGEVIN_SAMPLING( ... )

# 3. Compute energy of fake samples
E_fake := E_theta(x_fake)          # vector of size B

# 4. Data term of the loss:
- # push energies of real data down
- # push energies of fake data up
data_term := mean(E_real) - mean(E_fake)

# 5. Regularization on the magnitude of energies
# (to avoid energies growing to very large values)
reg_term := lambda * ( mean(E_real^2) + mean(E_fake^2) )

# 6. Total loss
loss := data_term + reg_term

```

$E_{\theta}(x)$ : تابع انرژی مدل که برای هر ورودی یک مقدار عددی تولید می‌کند.

$x\_real$ : یک mini-batch از داده‌های واقعی که مدل باید ساختار آن را یاد بگیرد.

Langevin Sampling: همان روش Langevin Sampling است که برای تولید ورودی مصنوعی از گرادینان انرژی و نویز استفاده می‌کند.

$x\_fake$ : داده‌های مصنوعی تولیدشده توسط Langevin برای مقایسه با داده‌های واقعی.

$E\_real$ : انرژی محاسبه‌شده برای داده‌های واقعی  $x\_real$ .

$E\_fake$ : انرژی محاسبه‌شده برای داده‌های مصنوعی  $x\_fake$ .

data term: بخشی از loss که تفاوت رفتار مدل روی  $real$  و  $fake$  را اندازه‌گیری می‌کند.

regularization term: جمله‌ای برای جلوگیری از بزرگ شدن نامطلوب مقدار انرژی‌ها.

loss: مجموع data term و regularization term که مدل آن را کمینه می‌کند.

تابع‌های خطای استفاده شده در آموزش:

$$\mathcal{L}_{data} = \frac{1}{B} \sum_{i=1}^B E_{real}^{(i)} - \frac{1}{B} \sum_{i=1}^B E_{fake}^{(i)}$$

$$\mathcal{L}_{reg} = \lambda \left( \frac{1}{B} \sum_{i=1}^B \left( E_{real}^{(i)} \right)^2 + \frac{1}{B} \sum_{i=1}^B \left( E_{fake}^{(i)} \right)^2 \right)$$

$$\mathcal{L} = \mathcal{L}_{data} + \mathcal{L}_{reg}$$

B : Batch size



---

### زیربخش سوم: تولید تصویر (۱۰ نمره)

از Langevin sampling استفاده کنید که از یک نویز Uniform بین  $[0,1]$  شروع کند و به تعداد کافی گام نمونه‌برداری انجام دهد. این کار را برای یک تعداد بار مثلاً ۱۶ بار انجام داده تا ۱۶ تصویر تولید شود و سپس ۱۶ تصویر نیز از داده‌های train بردارید و ۱۶ تا داده تولید شده و ۱۶ داده اصلی را چاپ کنید.

در گزارش:

- نتیجه به دست آمده را قرار دهید و بگویید داده‌های تولید شده چه کیفیتی دارند و اگر نتیجه خوب نیست علت اصلی خوب نشدن نتیجه را بیان کنید. (انجام بخش بعدی (نویززدایی تصویر نویزی) نیز می‌تواند در رسیدن به نتیجه‌گیری بهتر برای این بخش کمک کند).

---

### زیربخش چهارم: نویززدایی تصویر نویزی (۱۵ نمره)

۱. یک batch کوچک از تصاویر test بردارید (مثلاً ۱۶ تصویر).
۲. به این تصاویر نویز گاوسی اضافه کنید. (با noise level هایی که در انتها گفته می‌شود).
۳. تصاویر noisy را روی  $[0,1]$  clamp کنید و نشان دهید (در کنار تصاویر اصلی).
۴. از Langevin sampling استفاده کنید که از همین تصاویر noisy به عنوان ورودی اولیه شروع کند و به تعداد کافی گام نمونه‌برداری انجام دهد.
۵. تصاویر خروجی را نمایش دهید. (در کنار تصاویر real و noisy)
۶. گام‌های ۱ تا ۵ را برای noise level با سه مقدار ۰.۲ و ۰.۴ و ۰.۶ انجام دهید.

در گزارش:

- توضیح دهید که تا چه حد تصویر نویزی به تصویر اصلی نزدیک شده است.
- از بین مقادیر نویز گفته شده مشخص کنید که مدل کجا موفق بوده، کجا شکست خورده است.

### پی نوشت:

- همه نتایج (مانند عکس، خروجی کد و ...) و توضیحاتی که می‌خواهید بررسی و تصحیح شوند باید در گزارش آورده شوند و فقط به جواب‌های داخل گزارش نمره داده می‌شود. (بررسی کد جدا صورت می‌گیرد).
- اگر برای پاسخ به سوالی جوابی داده‌اید که حس می‌کنید ممکن است جواب شما با جواب طراح سوال متفاوت باشد، پیشنهاد می‌شود منبعی که جواب را از آن آورده‌اید به صورت دقیق (با دقت یک صفحه یا چند پاراگراف) ذکر کنید. مثلاً مقاله فلان صفحه فلان. این کار برای این است که اگر جواب در ذهن طراح سوال متفاوت با جواب شما بود ولی جواب شما هم جواب درستی بود نمره‌ای از دست ندهید.

## سوال دوم: SCORE-BASED MODELS

در سال‌های اخیر، مدل‌های مبتنی بر امتیاز به عنوان رویکردی قدرتمند و متمایز در یادگیری عمیق مولد ظهور کرده‌اند. برخلاف مدل‌های سنتی که سعی در تخمین مستقیم تابع چگالی احتمال دارند، این مدل‌ها بر یادگیری میدان برداری گرادیان لگاریتم تابع چگالی احتمال<sup>۴</sup> تمرکز می‌کنند. این تغییر استراتژی، محدودیت‌های محاسباتی مربوط به نرمال‌سازی توزیع‌ها را از میان برمی‌دارد و چارچوبی انعطاف‌پذیر برای تولید داده ارائه می‌دهد.

---

<sup>۴</sup> Score Function

## بخش اول: سوالات تئوری

### زیربخش اول (۵ نمره)

تابع امتیاز برای یک توزیع احتمال  $p(x)$  را تعریف کنید. فرض کنید یک مدل انرژی‌محور به صورت  $p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z(\theta)}$  داریم. نشان دهید که تابع امتیاز این مدل مستقل از تابع پارتیشن  $Z(\theta)$  است. این ویژگی چه مزیتی در آموزش مدل ایجاد می‌کند؟

### زیربخش دوم (۶ نمره)

تابع هزینه اصلی Score Matching شامل ترم  $tr(\nabla_x s_\theta(x))$  است. توضیح دهید چرا محاسبه این ترم برای داده‌های با ابعاد بالا (مانند تصاویر) دشوار و پرهزینه است. همچنین ایده اصلی روش  $DSM^5$  را توضیح دهید. همچنین ثابت کنید که مینیمم کردن تابع هدف DSM معادل مینیمم کردن تابع هدف Score Matching اصلی روی توزیع نویزی‌شده است (نیازی به اثبات دقیق ریاضی نیست، بیان منطق و رابطه بین گرادیان‌ها کافیست).

### زیربخش سوم (۴ نمره)

فرض کنید توزیع داده‌ها ترکیبی از دو توزیع گوسی با وزن‌های متفاوت باشد که تکیه‌گاه آن‌ها کاملاً از هم جداست (Disjoint Supports):

$$p_{data}(x) = \pi p_1(x) + (1 - \pi) p_2(x)$$

که در آن  $A$  و  $B$  نواحی مجزا هستند به طوری که  $p_1$  فقط در  $A$  و  $p_2$  فقط در  $B$  مقدار غیر صفر دارند. الف) نشان دهید که تابع امتیاز واقعی  $\nabla_x \log p_{data}(x)$  در هر ناحیه، هیچ وابستگی‌ای به ضریب مخلوط  $\pi$  ندارد. ب) توضیح دهید چرا این ویژگی باعث می‌شود الگوریتم Langevin Dynamics نتواند نسبت‌های صحیح وزن‌ها ( $\pi$  و  $1 - \pi$ ) را هنگام نمونه‌برداری رعایت کند و عملاً در نمونه‌برداری دچار مشکل شود.

## زیربخش چهارم (۵ نمره)

الف) مشکلاتی که هنگام اعمال مستقیم Score Matching روی داده‌های واقعی (بدون نویز) پیش می‌آید را نام ببرید و توضیح دهید چرا این مشکلات مانع یادگیری صحیح می‌شوند.

ب) با مراجعه به [مقاله](#) مدل‌های<sup>۶</sup> NCSN چگونه با استفاده از اغتشاش نویز چندمقیاسه<sup>۷</sup> و روش Annealed Langevin Dynamics این مشکلات را حل می‌کنند؟ نقش  $\sigma_1$  (بزرگترین نویز) و  $\sigma_L$  (کوچکترین نویز) را در کیفیت نمونه‌برداری توضیح دهید.

## بخش دوم: سوالات پیاده سازی

در این قسمت، شما یک مدل NCSN را پیاده‌سازی خواهید کرد. هدف نهایی، یادگیری میدان گرادیان چگالی احتمال و تولید تصاویر با کیفیت از ارقام دست‌نویس با استفاده از روش Annealed Langevin Dynamics است.

این سوال شامل دو بخش است: ابتدا مدل پایه را پیاده‌سازی کرده و سپس آن را به یک مدل شرطی ارتقا می‌دهید. تمام پارامترهای مورد نیاز برای آموزش پیشنهادی در انتهای سوال آورده شده‌اند.

## زیربخش اول: مدل پایه (۲۰ نمره)

۱. آماده سازی داده‌ها:

دیتاست [MNIST](#) را دانلود کرده و تصاویر را به بازه  $[-1, 1]$  نرمالایز کنید. سپس یک دنباله هندسی از سطوح نویز  $\{\sigma_i\}_{i=1}^L$  ایجاد کنید.

۲. معماری شبکه:

برای دستیابی به کیفیت مطلوب، از یک ساختار U-Net استفاده می‌کنیم که بلوک‌های اصلی آن AdaptiveResBlock هستند.

<sup>۶</sup> Noise Conditional Score Networks  
<sup>۷</sup> Multi-scale Noise Perturbation

این بلوک از مکانیزم **FiLM**<sup>۸</sup> برای تزریق اطلاعات نویز به شبکه استفاده می‌کند. به جای جمع ساده‌ی امبدینگ با فیچرها، امبدینگ نویز از طریق یک لایه Dense به دو پارامتر scale و shift تبدیل شده و سپس روی خروجی نرمال‌سازی اعمال می‌شود:

$$h = \text{GroupNorm}(h)$$

$$h = h \times (1 + \text{scale}) + \text{shift}$$

جدول ۲ معماری پیشنهادی شبکه را نشان می‌دهد، همچنین توجه کنید برای نرمال‌سازی خروجی نهایی شبکه را بر  $\sigma$  ورودی تقسیم کنید.

جدول ۲: معماری پیشنهادی شبکه ScoreNet

Stage	Layer / Block Type	Input Channels	Output Channels
Embedding	GaussianFourierProjection + Dense	Scalar ( $\sigma$ )	256
Input	Conv2d (3x3)	1	64
Encoder 1	AdaptiveResBlock	64	64
Down 1	AvgPool2d (2x2)	64	64
Encoder 2	AdaptiveResBlock	64	128
Down 2	AvgPool2d (2x2)	128	128
Encoder 3	AdaptiveResBlock	128	256
Up 1	Interpolate + Concat	256	384
Decoder 1	AdaptiveResBlock	384	128
Up 2	Interpolate + Concat	128	192
Decoder 2	AdaptiveResBlock	192	64
Output	Conv2d (3x3)	64	1

<sup>8</sup> Feature-wise Linear Modulation

### ۳. آموزش و خروجی‌ها:

مدل را با تابع هزینه Weighted Denoising Score Matching آموزش دهید و سپس الگوریتم نمونه‌برداری Annealed Langevin Dynamics را پیاده‌سازی کنید.

- نمودار خطای آموزشی را بر حسب ایپاک رسم کنید.
- یک تصویر Grid شامل ۱۶ رقم تولید شده نهایی نمایش دهید.
- برای ۳ نمونه مختلف، روند تبدیل شدن نویز خالص به عدد نهایی را در طی مراحل کاهش نویز  $(\sigma_{\text{begin}} \rightarrow \sigma_{\text{end}})$  به تصویر بکشید.
- نتایج را تحلیل کنید.

---

### زیربخش دوم: مدل شرطی (۱۵ نمره)

در مدل‌های مولد، اغلب نیاز داریم که فرآیند تولید را کنترل کنیم (مثلاً از مدل بخواهیم فقط عدد ۵ را تولید کند). برای این کار، مدل باید علاوه بر ساختار کلی داده، توزیع شرطی  $p(x|y)$  را نیز یاد بگیرد. برای تکنیک‌های آموزش این مدل‌ها به [مقاله](#) مراجعه فرمایید.

تغییرات لازم برای پیاده‌سازی:

- یک لایه nn.Embedding برای ۱۰ کلاس (ارقام ۰ تا ۹) به مدل اضافه کنید.
- در ورودی بلوک‌های AdaptiveResBlock، بردار امبدینگ نویز و بردار امبدینگ کلاس را با هم ترکیب کنید.
- مدل را مجدداً با استفاده از لیبل‌های دیتاست آموزش دهید.

در انتها:

- نمودار خطای آموزشی را بر حسب ایپاک رسم کنید.
- یک تصویر Grid تولید کنید که در آن هر ردیف مختص یک عدد خاص باشد (۰ تا ۹)
- نتایج این بخش را با بخش قبل مقایسه کنید و تحلیلی بر نتایج داشته باشید.

جدول ۳: مقادیر پیشنهادی هایپرپارامترها

<i>Parameter</i>	<i>Value</i>	<i>Description</i>
$\sigma \text{ begin}$	30	Initial noise level
$\sigma \text{ end}$	0.01	Final noise level
<i>Num Levels (L)</i>	10	Number of noise scales
<i>Batch Size</i>	64	-
<i>Optimizer</i>	Adam	-
<i>Learning Rate</i>	2e-4	-
<i>Epochs</i>	30	Minimum recommended
<i>Langevin Steps (T)</i>	150	Steps per noise level
<i>Step LR (<math>\epsilon</math>)</i>	2e-5	Langevin step size



## نکات تحویل

- مهلت ارسال این تمرین تا پایان روز "پنجشنبه ۴ دی ماه" خواهد بود.
- این زمان قابل تمدید نیست و در صورت نیاز می‌توانید از **grace time** استفاده کنید.
- در نظر داشته باشید که حداکثر مهلت آپلود تمرین در سامانه تا ۷ روز پس مهلت تحویل است و پس از آن سامانه بسته خواهد شد.
- پیاده سازی با زبان برنامه نویسی پایتون باید باشد و کدهای شما باید قابل اجرا بوده و به همراه گزارش آپلود شوند.
- انجام این تمرین به صورت یک نفره می‌باشد.
- در صورت مشاهده هر گونه تشابه در گزارش کار یا کدهای پیاده‌سازی، این امر به منزله تقلب برای طرفین در نظر گرفته خواهد شد.
- استفاده از کدهای آماده بدون ذکر منبع و بدون تغییر به منزله تقلب خواهد بود و نمره تمرین شما صفر در نظر گرفته می‌شود
- در صورت رعایت نکردن فرمت گزارش کار نمره گزارش به شما تعلق نخواهد گرفت.
- تحویل تمرین به صورت **دستنویس** قابل پذیرش نیست.
- تمامی تصاویر و جداول مورد استفاده در گزارش کار باید دارای توضیح (caption) و شماره باشند.
- بخش زیادی از نمره شما مربوط به گزارش کار و روند حل مسئله است.
- لطفا گزارش، فایل کدها و سایر ضمیمات مورد نیاز را با فرمت زیر در سامانه بارگذاری نمایید.
- HW3\_[Lastname]\_[StudentNumber].zip
- در صورت وجود سوال و یا ابهام می‌توانید از طریق رایانامه زیر با موضوع TAI\_HW3 با دستیاران آموزشی در ارتباط باشید:

○ سوال اول

[sinaprocomp@gmail.com](mailto:sinaprocomp@gmail.com)

○ سوال دوم

[farhadnasri999@gmail.com](mailto:farhadnasri999@gmail.com)

با آرزوی سلامتی و موفقیت روزافزون