Problem 1: (1+6+1=8 points total) Use the dataset – Movie Reviews dataset from kaggle (https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews) to identify the sentiments. From this, you can split and create training, validation and test data. Use training and validation data for training the model and test data for evaluation. Minimum iteration 50. You may apply early stopping. Build the sentiment analysis model:

a. Implement text preprocessing steps. You may apply various preprocessing techniques based on your task needs, such as tokenization, removing stopwords, stripping HTML, converting to lowercase, and lemmatization/stemming. You may include other preprocessing steps if necessary.

b. Utilize a combination of Word2Vec and Sequential models: RNN, LSTM, GRU, and BiLSTM.

- Finally, provide a table that includes the results of all combinations.

Problem 2 (6+2+2+2 =12 pts): Apply Deep Convolutional Generative Adversarial Networks (DCGAN) on any one of the following datasets: PathMnist, BloodMNIST OrganAMNIST, OrganCMNIST, and OrganSMNIST. For instance:

```
!pip install medmnist
from medmnist import PathMNIST
train_dataset = PathMNIST(split="train")
```

Number of **epochs should be minimum of 1000**. The architecture of the generator should follow the below architecture; however, you may change only the final convolutional layer to match with dimensions of the input real images. Discriminator architecture is open ended; however, you must include minimum of 3 convolutional layers.
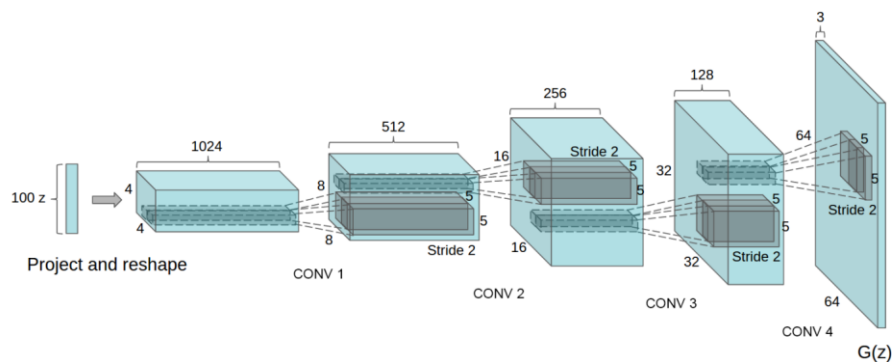
- Must present the learning curve including generator and discriminator loss and discuss presence of mode collapse issue
- Calculate FID score: use minimum 1000 real and 1000 generated images. Can you determine mode collapse based on the FID score? Explain.

- After completing the training process, generate 32 images and mention whether there is any issue of **mode collapse** in terms nature of generated images and generator and discriminator loss curve.

You are required to submit:

1. An MS/PDF/Scanned document:
    a. Include all the steps of your calculations.
    b. Attach screenshots of the code output.
    c. Include the summary of the model
    d. Include a Table - Mention all the hyperparameters you selected: activation function in hidden layer and output layer, weight initializer, number of hidden layers, neurons in hidden layers, loss function, optimizer, number of epochs, batch size, learning rate, evaluation metric
2. Source code:
    a. Python (Jupyter Notebook)
    b. Ensure it is well-organized with comments and proper indentation.
- Failure to submit the source code will result in a deduction of 5 points.
- Format your filenames as follows: "your_last_name_HW1.pdf" for the document and "your_last_name_HW1_source_code.ipynb" for the source code.
- Before submitting the source code, please double-check that it runs without any errors.
- Must submit the files separately.
- Do not compress into a zip file.
- HW submitted more than 24 hours late will not be accepted for credit.