Name: **Student ID**:

Quiz 11

1. The self-attention mechanism allows the inputs to interact with each other "self" and find out who they should pay more attention to "attention". The outputs are aggregates of these interactions and attention scores.

   Please sketch the self-attention block by explaining all functions of the following steps:

   (a) Prepare inputs, said Input 1, Input 2, and Input 3

   (b) Initialise weights

   (c) Derive key, query and value

   (d) Calculate attention scores for Input 1

   (e) Calculate softmax

   (f) Multiply scores with values

   (g) Sum weighted values to get Output 1

   (h) Repeat steps (d)–(g) for Input 2 & Input 3

   Hint. Please refer p.60 of Lecture 12.

2. Please sketch the masked self-attention block.

   Hint.

   Please refer p.71 of Lecture 12.

3. Please sketch the multihead self-attention block.

   Hint.

   Please refer p.72 of Lecture 12.

   Multi-headed attention improves the performance of the attention layer in two ways:

   (a) It expands the model's ability to focus on different positions. Yes, in the example above, z1 contains a little bit of every other encoding, but it could be dominated by the the actual word itself. It would be useful if we're translating a sentence like "The animal didn't cross the street because it was too tired", we would want to know which word "it" refers to.

   (b) It gives the attention layer multiple "representation subspaces". With multi-headed attention we have not only one, but multiple sets of Query/Key/Value weight matrices. Each of these sets is randomly initialized. Then, after training, each set is used to project the input embeddings (or vectors from lower encoders/decoders) into a different representation subspace.

4. Please sketch the Tranformer.

   Hint. The Transformer was proposed in the paper Attention Is All You Need.

   Harvard's NLP group created a guide annotating the paper with PyTorch implementation. In case you meet problems for HW12, you can refer the above link.