Home Work 8 - Scaled dot Product Attention

In [ ]:
```python
import torch; from torch import nn; from torch.nn import functional as F

d = 4; B = 1; T = 5
Q, K, V = torch.randn(B, T, d), torch.randn(B, T, d), torch.randn(B, T, d)

# PyTorch way
MHA = nn.MultiheadAttention(d, num_heads=1, bias=False, batch_first=True)
Wi, Wo = MHA.in_proj_weight, MHA.out_proj.weight
MHA_output, MHA_attention = MHA(Q, K, V) # shapes B, T, d and B, T, T

# Manual way
Wi_q, Wi_k, Wi_v = Wi.chunk(3)
Q, K, V = Q.squeeze(0), K.squeeze(0), V.squeeze(0) # remove batch dim

# write code here to derive `manual_attention` and `manual_output`.

# Apply projection weights
Q_proj = Q @ Wi_q.T
K_proj = K @ Wi_k.T
V_proj = V @ Wi_v.T

# Compute attention scores
scaled_dot_product = Q_proj @ K_proj.T / d**0.5
manual_attention = F.softmax(scaled_dot_product, dim=-1)

# Compute manual output
manual_output = manual_attention @ V_proj @ Wo.T

# Compare the two
print("Output:")
print(torch.allclose(MHA_attention, manual_attention)) # Aim for True
print(torch.allclose(MHA_output, manual_output)) # Aim for True
```

```
Output:
True
True
```