Problem Set4 Yuanda Wang
N:11865056

I construct an inverse maping and add an output node to the gModule to retrieve the predictions.
I have set vocab_size = 50 , seq_length = 0.5 and dropout = 0.5 and get the base line validation perplexity, which is 1276.

Q1.

```
input = nn.Identity()()
x3 = nn.Tanh()(nn.Linear(10, 20)(input))
x2 =  nn.Identity()()
x1 =  nn.Identity()()
X2 = nn.CMulTable()({x2, x3})
a = nn.CAddTable()({x1, X2})
m = nn.gModule({x1, x2, input}, {a})
```

Q2.
prev_c is previous memory cell.
$prev\_c = c_{t-1}^{l}$
prev_h is previous hidden state.
$prev\_h = h_{t-1}^{l}$

i is the result of D(h_t^{l-1}). Its output is scaled by 1/(1-dropout) or 0 directly.

Q3.
It returns the module in the format of CUDA, which is a parallel computing platform and programming model. It can be implemented by the GPUs. CUDA gives developers direct access to the virtual instruction set and memory of the parallel computational elements in CUDA GPUs. It is unrolled.

Q4.
model.s is a state in the neural network.
model.ds is the gradient of a state in the neural network.
model.start_s is the initialized state in the neural network.
We reset model.start_s when each layer begins during constructing the network. We also reset it after finishing dealing with the whole read-in file for a epoch.

Q5.
We clip the norm of the gradient at a certain number.

Q6.
Gradient descent

Q7.
I should add the Extra gradients to the tmp in the bp.