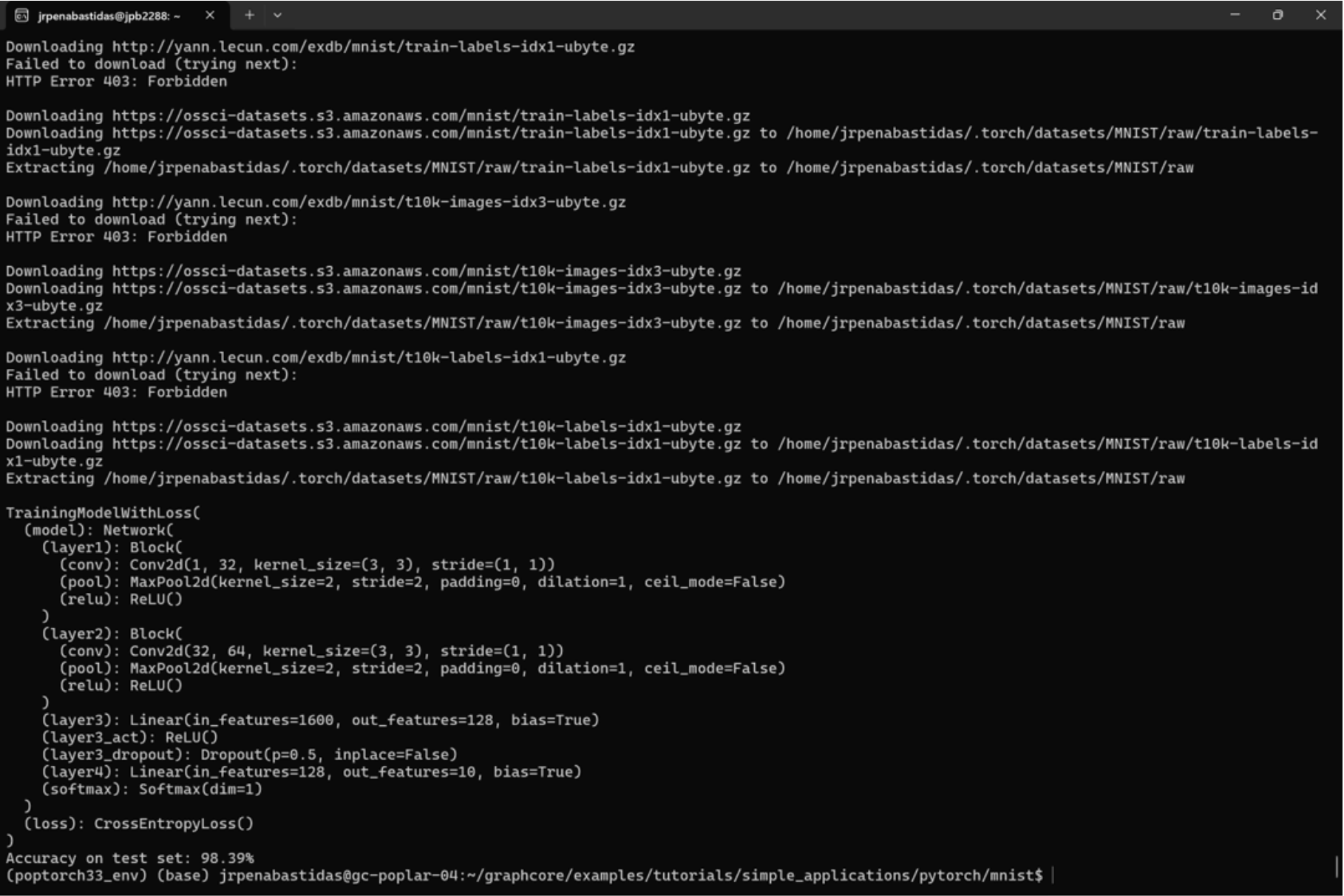
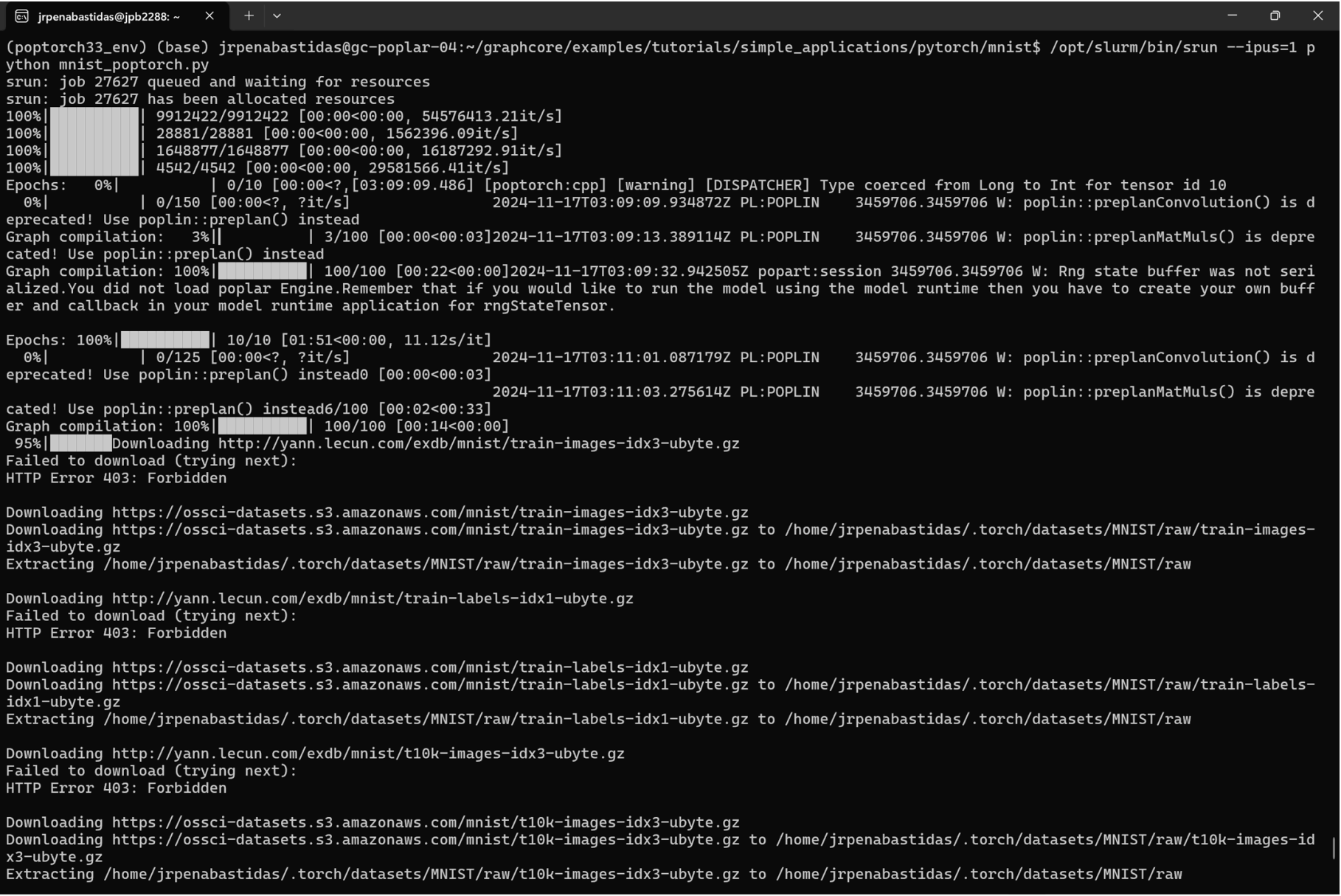


Homework

Using the Graphcore testbed

In []: *# Initial hyperparameters for the model*

```
learning_rate = 0.03
epochs = 10
batch_size = 8
test_batch_size = 80
```

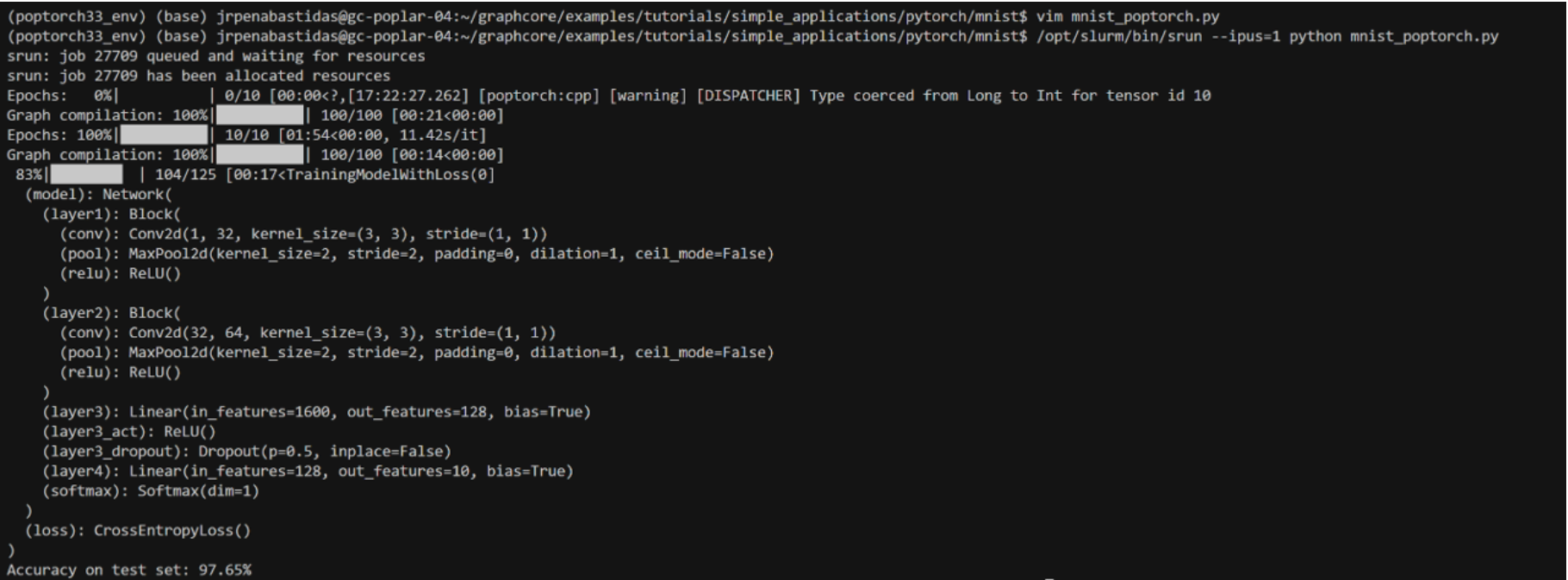


This is the baseline output for comparison. It took longer time to run as some downloads are required. The accuracy is 98.39% with this set of parameters.

First modification

```
In [ ]: # First modification to the hyperparameters

learning_rate = 0.1
epochs = 15
batch_size = 80
test_batch_size = 80
```

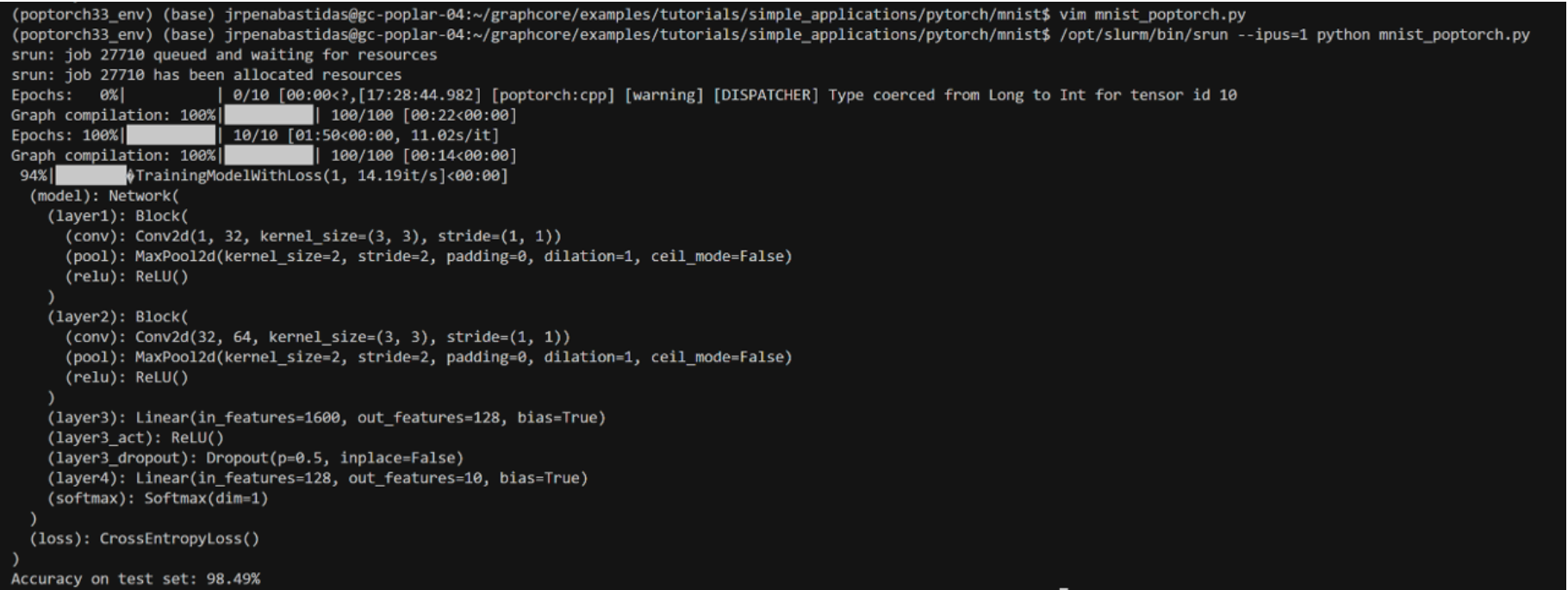


With bigger learning rate the accuracy is reduced to 97.65% even with a higher number of epochs

Second modification

```
In [ ]: # Second modification to the hyperparameters

learning_rate = 0.001
epochs = 20
batch_size = 8
test_batch_size = 80
```



With a smaller learning rate the accuracy returns to values above 98 %. Using lower number of epochs

Third modification

```
In [ ]: # Third modification to the hyperparameters

learning_rate = 0.0001
epochs = 20
batch_size = 16
test_batch_size = 80
```

```
(poptorch33_env) (base) jrpenabastidas@gc-poplar-04:~/graphcore/examples/tutorials/simple_applications/pytorch/mnist$ vim mnist_poptorch.py
(poptorch33_env) (base) jrpenabastidas@gc-poplar-04:~/graphcore/examples/tutorials/simple_applications/pytorch/mnist$ /opt/slurm/bin/srun --ipus=1 python mnist_poptorch.py
srun: job 27712 queued and waiting for resources
srun: job 27712 has been allocated resources
Epochs:   0%|          | 0/10 [00:00<?,[17:35:58.243] [poptorch:cpp] [warning] [DISPATCHER] Type coerced from Long to Int for tensor id 10
Graph compilation: 100%|          | 100/100 [00:20<00:00]
Epochs: 100%|          | 10/10 [01:51<00:00, 11.14s/it]
Graph compilation: 100%|          | 100/100 [00:14<00:00]
85%|          | 106/125 [00:17<00:00TrainingModelWithLoss(
(model): Network(
  (layer1): Block(
    (conv): Conv2d(1, 32, kernel_size=(3, 3), stride=(1, 1))
    (pool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (relu): ReLU()
  )
  (layer2): Block(
    (conv): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1))
    (pool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (relu): ReLU()
  )
  (layer3): Linear(in_features=1600, out_features=128, bias=True)
  (layer3_act): ReLU()
  (layer3_dropout): Dropout(p=0.5, inplace=False)
  (layer4): Linear(in_features=128, out_features=10, bias=True)
  (softmax): Softmax(dim=1)
)
  (loss): CrossEntropyLoss()
)
Accuracy on test set: 98.52%
```

In this case the learning rate is the smallest and a high number of epoch is used, the accuracy is high but not sustancially different from simulation less intensive in parameters.