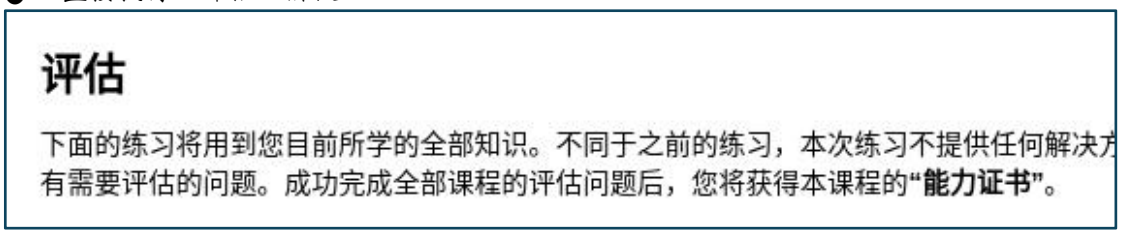
CUDA Python 通关引导

1. 基于 Numba 的 CUDA Python 编程简介

- 直接执行“评估”阶段：



- 需要修改的代码如下：（黄色块为添加或修改部分）

```
# Modify these 3 function calls to run on the GPU.
from math import exp
@vectorize(['float32(float32)'],target='cuda')
def normalize(grayscales):
    return grayscales / 255

@vectorize(['float32(float32,float32)'],target='cuda')
def weigh(values, weights):
    return values * weights

@vectorize(['float32(float32)'],target='cuda')
def activate(values):
    return ( exp(values) - exp(-values) ) / ( exp(values) + exp(-values) )
================== 下一格 ================
# Modify the body of this function to optimize .....
# As a constraint, even after you move work to the GPU,...
def  create_hidden_layer(n,  greyscales,  weights,  exp,  normalize,  weigh,
activate):
    #将 greyscales 和 weihts 在 GPU 上创建存储空间
    d_greyscales = cuda.to_device(greyscales)
    d_weights = cuda.to_device(weights)

    normalized = normalize(d_greyscales)
    weighted = weigh(normalized, d_weights)
    activated = activate(weighted)

    #将计算结果从 CPU 上传回 GPU 上
    activated_host = activated.copy_to_host()

    # The assessment mechanism will expect `activated` to be a host。。。
    # even after you refactor this code to run on the GPU, 。。。
    # `activated` back to the host.
    return activated_host
==================================
```

- 然后往下持续运行，执行时间小于 1 秒就可以：

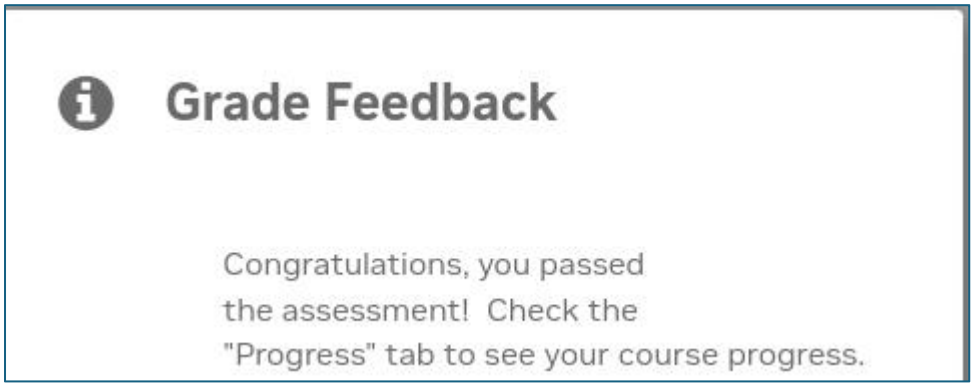
```
from assessment import assess
```

```
assess(create_hidden_layer, arguments)
```

```
Setting n to 100 million.

Your function returns a host np.ndarray: True

Your function took 0.47s to run.
Your function runs fast enough (less than 1 second): True

Your function returns the correct results: True
Congratulations, you passed! See the instructions below for how to get credit for your work to count toward a certi
ficate in the course.
```

- 到前面点击 ACCESS TASK

- 出现下面信息就表示通过



2. 使用 Numba 在 Python 中编写自定义 CUDA 核函数

- 在”**编写加速直方图核函数**”下面第三格“def cuda_histogram”上面有一段话：“.....请将此单元的内容粘贴至 assessment/histogram.py 并保存.....”，
- 点击 **assessment/histogram.py** 的链接，会开启另一个网页，
- 将下面代码贴入 histogram.py

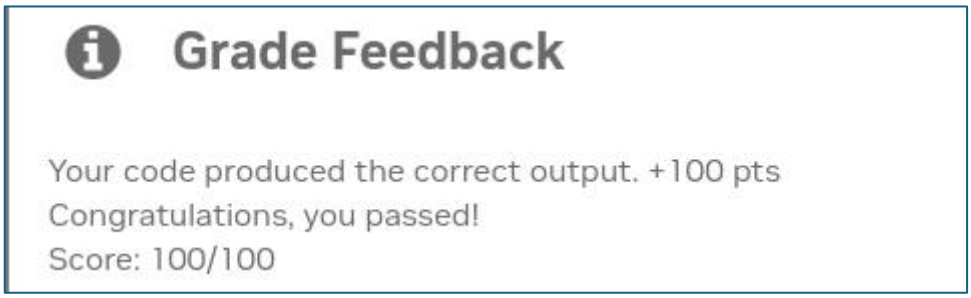
```
@cuda.jit
def cuda_histogram(x, xmin, xmax, histogram_out):
    nbins = histogram_out.shape[0]
    bin_width = (xmax - xmin) / nbins

    start = cuda.grid(1)
    stride=cuda.gridsize(1)

    for i in range(start, x.shape[0], stride):
        bin_number = np.int32((x[i] - xmin)/bin_width)
        if bin_number >= 0 and bin_number < histogram_out.shape[0]:
            ##主要修改这里用原子函数就好
            cuda.atomic.add(histogram_out, bin_number, 1)
```

最终结果如下图：

```
# Add your solution here
@cuda.jit
def cuda_histogram(x, xmin, xmax, histogram_out):
    nbins = histogram_out.shape[0]
    bin_width = (xmax - xmin) / nbins

    start = cuda.grid(1)
    stride=cuda.gridsize(1)


    for i in range(start, x.shape[0], stride):
        bin_number = np.int32((x[i] - xmin)/bin_width)
        if bin_number >= 0 and bin_number < histogram_out.shape[0]:
            ##主要修改这里用原子函数就好
            cuda.atomic.add(histogram_out, bin_number, 1)

```

- 点击左上角 File =》save 存储内容
- 到前面点击 ACCESS TASK

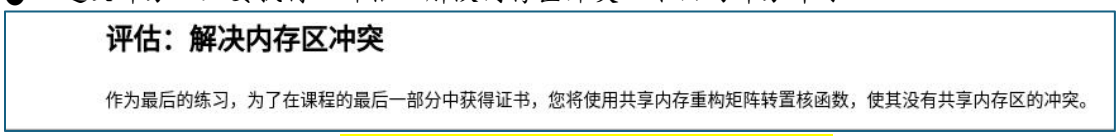
- 出现下面信息就表示通过



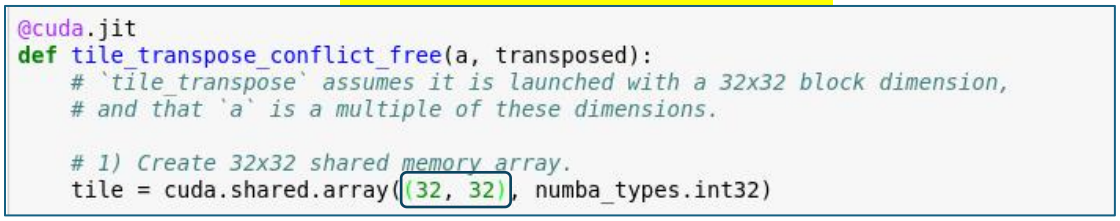
3. 有效使用内存子系统

- 里面有 3 处讲稿，请用“讲稿”进行搜索，然后执行该命令格，就会出现 PPT，然后可以下载

- 通关部分，只要执行“评估：解决内存区冲突”下面的部分即可：

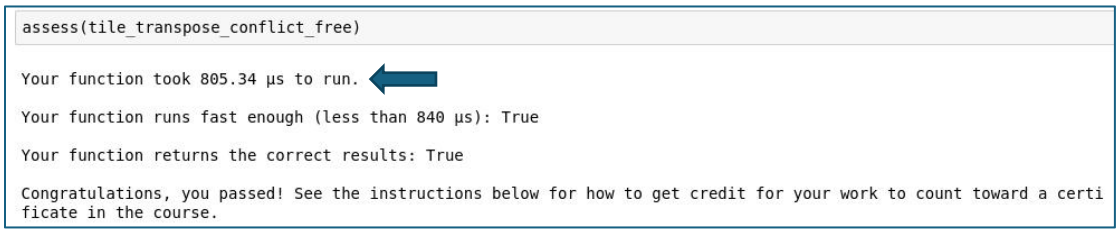


- 需要修改的部分如下：**将下图中 (32, 32) 改成 (33, 33) 即可**，然后执行到完



```
@cuda.jit
def tile_transpose_conflict_free(a, transposed):
    # `tile_transpose` assumes it is launched with a 32x32 block dimension,
    # and that `a` is a multiple of these dimensions.

    # 1) Create 32x32 shared memory array.
    tile = cuda.shared.array((32, 32), numba_types.int32)
```

- 最后执行时间小于 840us 即可



```
assess(tile_transpose_conflict_free)

Your function took 805.34 µs to run.

Your function runs fast enough (less than 840 µs): True

Your function returns the correct results: True

Congratulations, you passed! See the instructions below for how to get credit for your work to count toward a certi
ficate in the course.
```

- 通过之后，别忘了到前面执行“ASSESS TASK”

- 如果全部都通过的话，就会看到下面信息：

Grade Feedback

Congratulations, you passed the assessment! Check the "Progress" tab to see your course progress. After you have completed the assessment in all 3 tasks, click the "View Certificate" button to receive your certificate for the workshop.