1. Consider the following statements regarding Artificial Neural Network s(ANN) and Convolutional Neural Networks (CNN)

```
     1.There are sparse connections between inputs and outputs
between two consecutive layers in a CNN
     2.Parameters are shared between output neurons in a CNN layer.
     3.There are sparse connections between inputs and outputs
between two consecutive layers of an ANN
     4.For any two layers with the same number of neurons an ANN
will have fewer parameters than a CNN

    Which of the above statements are TRUE
```

- **A. RIGHT** 1 and 2
- **B.** 1, 2 and 3
- **C.** 1, 3 and 4
- **D.** 2, 3 and 4

2. What will be the size of the output of a convolutional layer with :

```
Input size = [ 227 x 227 x 3 ],
Filter Size = [ 11 x 11 x 3],
Stride = 4
```

- A. [54 x 54]
- **B. RIGHT** [55 x 55]
- C. [216 x 216]
- D. [68 x 68]

$$O = \frac{(W - K + 2P)}{S} + 1$$

Here W = 227, K=11, P=0m and S = 4 for both height and width

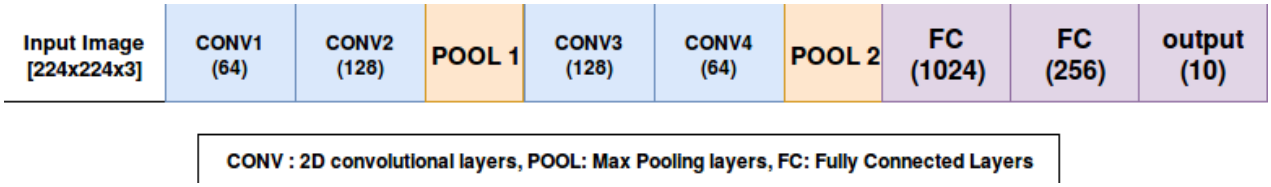
So O = (227-11+2*0)/4+1 = 55

3. Pooling layers are used to accomplish which of the following?

- **A. RIGHT** To progressively reduce the spatial size of the representation
- **B. RIGHT** To reduce the amount of parameters and computation in network
- C. To select maximum value over pooling region always
    - Selecting max value always is max pooling. but there are other operations also possible in pooling layer like average pooling.
- D. None of the above

Answer questions 4-6 for the CNN architecture given below

The whole network is composed of CONV layers that perform 3x3 convolutions with stride 1 and padding is 'same'. POOL layers perform 2x2 max pooling with stride 2 (and no padding). Number of filters in the Conv layers and number of neurons in fully connected layers are shown in brackets



4. The output size after pool1, pool2 are

- A. [111x111x128], [56x56x64]
- **B. RIGHT** [112x112x128], [56x56x64]
- C. [114x114x128], [58x58x64]
- D. [111x111x128], [58x58x64]

Each convolution layer will output same HxW since padding is SAME and stride is 1.

- So after CONV1 224x224x64
- So after CONV2 224x224x128 pooling size is 2x2 and stride is 2. It means no overlaping in inputspace.
- So after POOL1 **112x112x128**
- So after CONV3 112x112x128
- So after CONV4 112x112x64
- So after POOL2 56x56x64

5. Number of parameters till pool1 are

- **A**. 89186
- **B**. 73570
- **C RIGHT**. 75648
- **D**. 64898

Paramters in the Conv1

= no.of.filters *(filter-width* filter-height * input-depth + 1-bias-parameter)

= 64 *(3* 3 * 3 + 1) = 1792

Paramters in the Conv2

= no.of.filters *(filter-width* filter-height * input-depth + 1-bias-parameter)

= 128 *(3* 3 * 64 + 1) = 73856

Parameters till pool 1

= Paramters in the Conv1 + Paramters in the Conv2 = 1792 + 73856 = **75648**

6. Total number of parameters in the given network is

- **A**. 104097392
- **B**. **206081344**
- **C**. 326789108
- **D**. 207816190

My understanding is that the ouput width and height of all conv layers is same as of input width and height since stride 1 and padding 'same' is for all conv layers(https://stackoverflow.com/a/42195267/739331). The pooling is 2x2 and non-overlaping (since stride is 2). That means output of each pooling layer is input-width/2 x input-height/2 x input-depth.

INPUT LAYER

- input : N/A
- output: 224 *224* 3

CONV1 LAYER

- input : 224 *224* 3
- Number of filters = 64
- Filter size = 3 *3* 3 (depth of filter same as input depth)
- Parameters = Number of filters * (filter size + 1)

```
= 64 * (3 * 3 * 3 + 1) = 1792
```

- output: 224 *224* 64 (depth of output same as number of filters)

CONV2 LAYER

- input : 224 *224* 64
- Number of filters = 128
- Filter size = 3 *3* 64
- Parameters = 128 *(3* 3 * 64 + 1) = 73856
- output: 224 *224* 128

POOL1 LAYER

- input : 224 *224* 128
- Parameters = No parameters for any pool layers.
- output: 112 *112* 128

CONV3 LAYER

- input : 112 *112* 128
- Number of filters = 128
- Filter size = 3 *3* 128
- Parameters = 128 *(3* 3 * 128 + 1) = 147584
- output: 112 *112* 128

CONV4 LAYER

- input : 112 *112* 128
- Number of filters = 64
- Filter size = 3 *3* 128
- Parameters = 64 *(3* 3 * 128 + 1) = 73792
- output: 112 *112* 64

POOL2 LAYER

- input : 112 *112* 64
- output: 56 *56* 64

FC 1024 LAYER

- Neurons from previous layer: 56 *56* 64 = 200704
- Neurons int the current layer: 1024
- Parameters/weights = (Neurons from previous layer + 1-bias-neuron) * Neurons int the current layer.

```
= (200704 + 1) * 1024  = 205521920
```

- output: 1024 neurons

FC 256 LAYER

- Neurons from previous layer: 1024
- Neurons int the current layer: 256
- Parameters/weights = (1024 + 1) * 256 = 262400
- output: 256 neurons

OUTPUT LAYER

- Neurons from previous layer: 256
- Neurons int the current layer: 10
- Parameters/weights = (256 + 1) * 10 = 2570
- output: Never mind!

Parameters

- conv1: 1792
- conv2: 73856
- conv3: 147584
- conv4: 73792
- fc1024:205521920
- fc256: 262400
- output: 2570
- TOTAL: 206083914

Total parameters excluding output paramters = 206083914 - 2570 = 206081344 (this is one of the option!)

7. Which of the following is true for most CNN architectures?

- **A RIGHT**. Size of input (height and width) decreases, while depth increases
- **B RIGHT**. Multiple convolutional layers followed by pooling layers.
- **C**. Fully connected layers in the first few layers
- **D RIGHT**. Fully connected layers in the last few layers
- **E**. Multiple pool layers followed by a convolutional layer

Consider the architecture shown below and answer Questions 8-10

Type	path size/stride	output size	depth	#1X1	#3X3 reduce	#3X3	#5X5 reduce	#5X5	pool proj
convolution	7X7/2	112X112X64	1						
max pool	3X3/2	56X56X64	0						
convolution	3X3/1	56X56X192	2		64	192			
max pool	3X3/2	28X28X192	0						
(3a)		28X28X256	2	64	96	128	16	32	32

8. The network shown is popularly known as

- **A**. AlexNet
- **B**.VGG
- **C RIGHT**. GoogLeNet
    - CNN Architecture Part 3 (GoogleNet)
- **D**.ResNet

9. What are the number of parameters and number of operations, for layer (3a) in above question?

- **A**. #parameters=163 K (approx.), #operations=128 M (approx.)
- ~~**B RIGHT**~~ . #parameters=159 K (approx.), #operations=128 M (approx.)
    - CNN Architecture Part 3 (GoogleNet)
- **C**. #parameters=128 M (approx.), #operations=159 K (approx.)
- **D**. #parameters=128 K (approx.), #operations=159 M (approx.)

Accepted answer is **A**

---

10. The importance of “reduce” in the table is that it

- **A**. Reduces no. of feature maps in the previous layer
- **B**. Reduces no. of operations
- **C**. Reduces no. of parameters
- **D RIGHT**. All of the above

REFERNECES

1. The best explanation of Convolutional Neural Networks on the Internet!