def compute\_metrics(y\_true, y\_pred):

  '''

  Computes IOU and Dice Score.

  Args:

    y\_true (tensor) - ground truth label map

    y\_pred (tensor) - predicted label map

  '''

  class\_wise\_iou = []

  class\_wise\_dice\_score = []

  smoothening\_factor = 0.00001

  for i in range(12):

    intersection = np.sum((y\_pred == i) \* (y\_true == i))

    y\_true\_area = np.sum((y\_true == i))

    y\_pred\_area = np.sum((y\_pred == i))

    combined\_area = y\_true\_area + y\_pred\_area

    iou = (intersection + smoothening\_factor) / (combined\_area - intersection + smoothening\_factor)

    class\_wise\_iou.append(iou)

    dice\_score =  2 \* ((intersection + smoothening\_factor) / (combined\_area + smoothening\_factor))

    class\_wise\_dice\_score.append(dice\_score)

  return class\_wise\_iou, class\_wise\_dice\_score