```python
import numpy as np

from onnxruntime import InferenceSession

def main():
    onnxModelFilename = R'model.onnx'

    file = open(onnxModelFilename, "rb")
    onnxModelBytes = file.read()
    file.close()

    session = InferenceSession(onnxModelBytes)

    #Make Prediction
    print("[%i, %i] = %i" % (0, 0, Predict(session, 0, 0)))
    print("[%i, %i] = %i" % (0, 1, Predict(session, 0, 1)))
    print("[%i, %i] = %i" % (1, 0, Predict(session, 1, 0)))
    print("[%i, %i] = %i" % (1, 1, Predict(session, 1, 1)))

def Predict(session: InferenceSession, p: int, q: int) -> int:
    x = np.array([[[p, q]]], "float32")
    feed = dict([(sessionInput.name, x[n]) for n, sessionInput in enumerate(session.get_inputs())])

    prediction = session.run(None, feed)
    predictedValue = prediction[0].flatten()[0]
    roundedPredictedValue = int(round(predictedValue))

    return roundedPredictedValue

if __name__ == "__main__":
    main()

```