```
#Matematyka Konkretna

#Laboratorium 8

#Szymon Białek https://github.com/NynyNoo/MK

#Wariant 11

import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import SimpleRNN, Dense

def generate_data_mean(num_samples, seq_length):
    X = np.random.choice([0, 0.2, 0.4, 0.6, 0.8, 1], 
size=(num_samples, seq_length, 1))
    y = np.mean(X, axis=1)
    return X, y

num_samples = 30
seq_length = 20
input_dim = 1
output_dim = 1

X_train, y_train = generate_data_mean(num_samples, seq_length)

model = Sequential()
model.add(SimpleRNN(units=10, input_shape=(seq_length, input_dim)))
model.add(Dense(units=output_dim, activation='linear'))

model.compile(optimizer='adam', loss='mean_squared_error', 
metrics=['mae'])  # Używam średniego błędu bezwzględnego (mae) jako 
metryki.

model.fit(X_train, y_train, epochs=100, batch_size=1, verbose=2)

X_test, y_test = generate_data_mean(5, seq_length)
predictions = model.predict(X_test)

for i in range(len(X_test)):
    print("Input:", X_test[i].flatten())
    print("True Output:", y_test[i])
    print("Predicted Output:", predictions[i][0])
    print("\n")
```