```
#14 Write a program to implement RNN
from tensorflow.keras.datasets import imdb
```

```
(X_train,y_train),(X_test,y_test)=imdb.load_data(num_words=20000)
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/imdb.npz
17464789/17464789 [==============================] - 0s 0us/step
```

```
X_train.shape,X_test.shape
```

```
((25000,), (25000,))
```

```
len(X_train[0]),len(X_train[1]),len(X_train[2]),len(X_train[3]),len(X_train[4])
```

```
(218, 189, 141, 550, 147)
```

```
y_train[:5]
```

```
array([1, 0, 0, 1, 0])
```

```
X_train[0]
```

```
[1,
 14,
 22,
 16,
 43,
 530,
 973,
 1622,
 1385,
 65,
 458,
 4468,
 66,
 3941,
 4,
 173,
 36,
 256,
 5,
 25,
 100,
 43,
 838,
 112,
 50,
 670,
 2,
 9,
 35,
 480,
 284,
 5,
 150,
 4,
 172,
 112,
 167,
 2,
 336,
 385,
 39,
 4,
 172,
 4536,
 1111,
 17,
 546,
 38,
 13,
 447,
 4,
 192,
 50,
 16,
 6,
 147,
 2025,
 19,
```

```
import numpy as np
```

```
np.array(X_train[0])
```

```
array([    1,    14,    22,    16,    43,   530,   973,  1622,  1385,
          65,   458,  4468,    66,  3941,     4,   173,    36,   256,
           5,    25,   100,    43,   838,   112,    50,   670,     2,
           9,    35,   480,   284,     5,   150,     4,   172,   112,
         167,     2,   336,   385,    39,     4,   172,  4536,  1111,
          17,   546,    38,    13,   447,     4,   192,    50,    16,
           6,   147,  2025,    19,    14,    22,     4,  1920,  4613,
         469,     4,    22,    71,    87,    12,    16,    43,   530,
          38,    76,    15,    13,  1247,     4,    22,    17,   515,
          17,    12,    16,   626,    18, 19193,     5,    62,   386,
          12,     8,   316,     8,   106,     5,     4,  2223,  5244,
          16,   480,    66,  3785,    33,     4,   130,    12,    16,
          38,   619,     5,    25,   124,    51,    36,   135,    48,
          25,  1415,    33,     6,    22,    12,   215,    28,    77,
          52,     5,    14,   407,    16,    82, 10311,     8,     4,
         107,   117,  5952,    15,   256,     4,     2,     7,  3766,
           5,   723,    36,    71,    43,   530,   476,    26,   400,
         317,    46,     7,     4, 12118,  1029,    13,   104,    88,
           4,   381,    15,   297,    98,    32,  2071,    56,    26,
         141,     6,   194,  7486,    18,     4,   226,    22,    21,
         134,   476,    26,   480,     5,   144,    30,  5535,    18,
          51,    36,    28,   224,    92,    25,   104,     4,   226,
          65,    16,    38,  1334,    88,    12,    16,   283,     5,
          16,  4472,   113,   103,    32,    15,    16,  5345,    19,
         178,    32])
```

```
from tensorflow.keras.preprocessing.sequence import pad_sequences
```

```
X=pad_sequences(X_train,maxlen=200)
X_val=pad_sequences(X_test,maxlen=200)
```

```
len(X[0])
```

```
200
```

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM,Dense,Embedding
```

```
model=Sequential()
model.add(Embedding(20000,128,input_shape=(200,)))
model.add(LSTM(100,return_sequences=True))
model.add(LSTM(100))
model.add(Dense(1,activation='sigmoid'))
```

```
model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])
```

```
model.fit(X,y_train,validation_data=(X_val,y_test),epochs=5,batch_size=64)
```

```
Epoch 1/5
391/391 [==============================] - 351s 888ms/step - loss: 0.4098 - accuracy: 0.8132 - val_loss: 0.3552 - val_accuracy: 0.85
Epoch 2/5
391/391 [==============================] - 337s 862ms/step - loss: 0.2205 - accuracy: 0.9155 - val_loss: 0.3394 - val_accuracy: 0.85
Epoch 3/5
391/391 [==============================] - 339s 869ms/step - loss: 0.1483 - accuracy: 0.9462 - val_loss: 0.3804 - val_accuracy: 0.85
Epoch 4/5
391/391 [==============================] - 340s 869ms/step - loss: 0.1003 - accuracy: 0.9636 - val_loss: 0.4294 - val_accuracy: 0.84
Epoch 5/5
391/391 [==============================] - 340s 869ms/step - loss: 0.0738 - accuracy: 0.9738 - val_loss: 0.5268 - val_accuracy: 0.85
<keras.src.callbacks.History at 0x7be8988a2ce0>
```