#include <iostream>

#include <string>

#include <cstdlib>

#include <ctime>

#include <fstream>

using namespace std;

int main()

{

int layer\_num ;

/\*\*\*\*\*\*\*\*\*\* Get the layer number \*\*\*\*\*\*\*\*\*\*/

cout<<"Enter Layer # "<<endl;

cin >> layer\_num;

string WEIGHTS\_FILE\_NAME="weights"+to\_string(layer\_num)+".txt";

string IN\_FILE\_NAME="input\_layer"+to\_string(layer\_num)+".txt";

string OUT\_FILE\_NAME="output\_layer"+to\_string(layer\_num)+".txt";

string OUT\_FILE\_NAME2="input\_layer"+to\_string(layer\_num+1)+".txt";

int NUM\_INPUTS = (layer\_num==1)?100:(layer\_num==2)?32:10;

int NUM\_OUTPUTS = (layer\_num==1)?32:(layer\_num==2)?10:4;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\* Vectors X,Y and Matrix W \*\*\*\*\*\*\*\*/

float x[NUM\_INPUTS],y[NUM\_OUTPUTS],w[NUM\_INPUTS][NUM\_OUTPUTS];

/\*\*\*\*\*\*\* File stream constructors \*\*\*\*\*\*\*/

ifstream fin (WEIGHTS\_FILE\_NAME);

ofstream foutY(OUT\_FILE\_NAME);

ofstream foutY2(OUT\_FILE\_NAME2);

/\*\*\*\* For first layer - generate random numbers \*\*\*\*\*/

if(layer\_num==1)

{

srand((unsigned int)time(NULL));

//Generate random numbers for the inputs

ofstream fout(IN\_FILE\_NAME);

float maxFloat = 1;

for(int i=0;i<NUM\_INPUTS;i++)

{

x[i]= (float(rand())/float((RAND\_MAX)) \* maxFloat);

//save them at the output file

fout<<x[i]<<endl;

}

fout.close();

}/\*\*\*\* For subsequent layers - read the previous generated output \*\*\*\*\*\*\*/

else

{

ifstream fin\_X(IN\_FILE\_NAME);

for(int i=0;i<NUM\_INPUTS;i++)

{

fin\_X >> x[i];

}

fin\_X.close();

}

/\*\*\*\*\*\* Read the weights \*\*\*\*\*\*\*\*\*\*\*/

for(int i=0;i<NUM\_INPUTS;i++)

for(int j=0;j<NUM\_OUTPUTS;j++)

fin>>w[i][j];

/\*\*\*\*\*\*\*\*\* Compute the Layer Output \*\*\*\*\*\*\*\*/

//Compute the first layer output

for(int j=0;j<NUM\_OUTPUTS;j++)

{

y[j]=0;

for(int i=0;i<NUM\_INPUTS;i++)

y[j]+=x[i]\*w[i][j];

/\*\*\*\* Activation Function \*\*\*\*/

if(y[j]<0)

y[j]=0;

foutY<<y[j]<<endl;

foutY2<<y[j]<<endl;

}

foutY.close();

foutY2.close();

return 0;

}