

Project Euler Problem 11 Solution

```
#include <iostream>

using namespace std;

int main()
{

    // Will be 20 for problem
    int dimensions;
    cout << "Enter a size: ";
    cin >> dimensions;

    int numMatrix[dimensions][dimensions];

    // Reads in matrix
    for (int x = 0; x < dimensions; x++)
    {

        for (int y = 0; y < dimensions; y++)
        {
            cin >> numMatrix[x][y];
        }

    }

    int greatestProduct = 0;

    for (int y = 0; y < dimensions; y++)
    {

        for (int x = 0; x < dimensions; x++)
        {

            // Checks right
            if (x + 3 < dimensions)
            {

                int currentGreatest = numMatrix[x][y] * numMatrix[x+1][y] * numMatrix[x+2][y] *
numMatrix[x+3][y];
                if (greatestProduct < currentGreatest)
                {
                    greatestProduct = currentGreatest;
                }
            }
        }
    }
}
```

Project Euler Problem 11 Solution

```
}

// Checks down
if (y + 3 < dimensions)
{

    int currentGreatest = numMatrix[x][y] * numMatrix[x][y+1] * numMatrix[x][y+2] *
numMatrix[x][y+3];
    if (greatestProduct < currentGreatest)
    {
        greatestProduct = currentGreatest;
    }

}

// Checks left-diagonal
if (x + 3 < dimensions && y + 3 < dimensions)
{

    int currentGreatest = numMatrix[x][y] * numMatrix[x+1][y+1] * numMatrix[x+2][y+2] *
numMatrix[x+3][y+3];
    if (greatestProduct < currentGreatest)
    {
        greatestProduct = currentGreatest;
    }

}

// Checks right-diagonal
if (x + 3 < dimensions && y >= 3)
{

    int currentGreatest = numMatrix[x][y] * numMatrix[x+1][y-1] * numMatrix[x+2][y-2] *
numMatrix[x+3][y-3];
    if (greatestProduct < currentGreatest)
    {
        greatestProduct = currentGreatest;
    }

}

}
```

Project Euler Problem 11 Solution

```
cout << "The answer is: " << greatestProduct << endl;  
  
}
```