```cpp
/****************************************************************************
*       Author:                 Jessica Schuler
*       Date Created:           11-21-13
*       Filename:               greatProd.cpp
*
*       Overview:
*               This program gets user command line input and fills an array
*               with random numbers.  The array will then be searched for
*               tetris shapes and the shape with the greatest product will
*               be identified.
*       Input:
*               The user inputs the desired number of rows and columns for
*               the randomly generated array.
*       Output:
*               The program will identify the location, shape, and product
*               of the shape with the greatest product.
****************************************************************************/

#include<iostream>
#include<ctime>
#include<cstdlib>

using namespace std;
void get_info(int &rows, int &cols);
void fill_array(int rows, int cols);

int main (int argc, char *argv[])
{
    int rows=0,cols=0;
    rows = atoi(argv[1]);
    cols = atoi(argv[2]);
    get_info(rows, cols);
        fill_array(rows, cols);

    return 0;
}

//Function to get user input and validate input
void get_info(int &rows, int &cols) {
        if(rows<4){
            cout<<"Enter a number 4 or higher rows!!";
            cin>>rows;}
             if(cols<4){
            cout<<"Enter a number 4 or higher for columns!!";
            cin>>cols;}}

//Function to fill the array with random numbers
void fill_array(int rows, int cols){
    srand(time(0));//Random seed generator for random #'s
    int** arr_grid = new int*[rows];//allocates a 2-d array
    for(int i = 0; i < rows; i++)  {
      arr_grid[i]= new int[cols]; }
    for(int i = 0; i < rows; i++) {  //fills the array w/random #'s
        for(int j = 0; j < cols; j++) {
            arr_grid[i][j]= rand()%99 + 0;}}
    for(int i=0; i<rows; i++) {    //output the array
            for(int j=0; j<cols; j++) {
            cout<<arr_grid[i][j]<<" "; }
        cout<<endl; }
    for(int i=0; i<rows; i++) { //delete the array
        delete [] arr_grid[i]; }
    delete [] arr_grid; }
```