Exercise on Arrays and Loops

1. Create a program that will ask the user to enter a string, and the program will print all the unique characters in the string.
2. Given an array that contains positive and negative integers, create a program that will determine the maximum product sub-array.

i.e.: {6,-3,-10,0,2} => maximum subproduct array: {6,-3,-10} (product = 180)

1. Create a program that will ask the users to enter two strings, and the program will print all the uncommon characters between the two strings.
2. Re-create the following programs without using any loops [convert to recursive programs]:

|  |
| --- |
| #include <iostream>  using namespace std;  int main()  {  int rows;  cout << "Enter number of rows: ";  cin >> rows;  for(int i = rows; i >= 1; --i)  {  for(int j = 1; j <= i; ++j)  {  cout << j << " ";  }  cout << endl;  }  return 0;  } |
| #include <iostream>  using namespace std;  int main()  {  int space, rows;  cout <<"Enter number of rows: ";  cin >> rows;  for(int i = 1, k = 0; i <= rows; ++i, k = 0)  {  for(space = 1; space <= rows-i; ++space)  {  cout <<" ";  }  while(k != 2\*i-1)  {  cout << "\* ";  ++k;  }  cout << endl;  }  return 0;  } |
| int main()  {  int rows, count = 0, count1 = 0, k = 0;  cout << "Enter number of rows: ";  cin >> rows;  for(int i = 1; i <= rows; ++i)  {  for(int space = 1; space <= rows-i; ++space)  {  cout << " ";  ++count;  }  while(k != 2\*i-1)  {  if (count <= rows-1)  {  cout << i+k << " ";  ++count;  }  else  {  ++count1;  cout << i+k-2\*count1 << " ";  }  ++k;  }  count1 = count = k = 0;  cout << endl;  }  return 0;  } |
| #include <iostream>  using namespace std;  int main()  {  int rows, number = 1;  cout << "Enter number of rows: ";  cin >> rows;  for(int i = 1; i <= rows; i++)  {  for(int j = 1; j <= i; ++j)  {  cout << number << " ";  ++number;  }  cout << endl;  }  return 0;  } |