Assignment: Program **p02** prints all the elements in the Cartesian product of two sets.

Prohibition: Use of the C++ Standard Template Library is prohibited in the implementation of this project.

Program Files: Project **2** consists of files **p02.cpp, List02.h, List02.cpp,** and **p02make.** Project 2 file names are exactly as given. Failure to employ the foregoing names will result in a score of ***zero*** (***0***) for this project

Project files must be stored in the **root directory of your student account**. Failure to store project files in the root directory of your student account will result in a score of ***zero*** (***0***) for this project.

File	**Description**
p02.cpp	File **p02.cpp** contains functions that process command line arguments and compute results specified in the assignment.
List02.h	File **List02.h** contains the interface of class *List* employed to represent a set from which values are drawn to make a Cartesian product.
List02.cpp	File **List02.cpp** contains the implementation of class *List*.
p02make	File **p02make** contains instructions that create executable program **p02**. File **p02make** is interpreted by the Linux command ***make***.

Display and Keyboard Input: In the dialog below, the user enters text shown in bold.

Command Line: Project **2** can be invoked with zero, one, two or three program parameters. The first program parameter is the name of the input file containing the first of the two sets from which the Cartesian product is created. The second parameter is the name of the second of the two sets from which the Cartesian product is created. The third parameter is the name of the file containing the set of Cartesian products, listed as ordered pairs. Sample command lines together with corresponding actions by program **p02** are shown below. Boldfaced type indicates data entered at the keyboard by the user.

```
$ p02
Enter the name of input file 1: i021.dat
Enter the name of input file 2: i022.dat
Enter the output file name: o02.dat

$ p02 i021.dat
Enter the name of input file 2: i022.dat
Enter the output file name: o02.dat

$ p02 i021.dat i022.dat
Enter the output file name: o02.dat

$ p02 i021.dat i022.dat o02.dat
```

Input File Specifications: The input file consists of a sequence of unique strings. No two strings can be identical. Strings are separated by white space. White space is one or more characters from the set that includes a blank character, a newline character, or a tab character.

Example Input File 1: a b c

Example Input File 2: 1 2

Output File Specification: The output file consists of the set of Cartesian products created from the two input files. Elements of the set are enclosed in curly braces, { }, and separated by commas. Each element of the set is an ordered pair enclosed in parentheses where an element from the first set is separated from and element from the second set by a comma.

Example Output File

```
L={a,b,c}
M={1,2}
L X M={(a,1),(a,2),(b,1),(b,2),(c,1),(c,2)}
```

```
struct ListException {
    ListException(const char* m)
    {   cout << endl;
        cout << “I am the List and I am “ << m << “.”;
        cout << endl;
    }
};
class List {
    int size;               //Number of available elements
    int count;              //Number of occupied elements
                            //Index of the next available element
    int cursor;             //Index of the current element on the list
    string* L;              //Points to an array of strings used to implement the
                            //List
public:
    List(int sz=100):size(sz),count(0),cursor(0){L=new string[size];}
    List(istream& i,int sz=100):size(sz),count(0),cursor(0)
    {   L=new string[size];
        Scan(i)
    }
    ~List(){if (L) delete[] L;}
    bool IsFull(void){return count>=size-1;}
    void Insert(const string& v){}
    void Print(ostream& o,const char* title){}
    void Scan(istream& i){}
    void First(void){]
    void Next(void){}
    bool IsEol(void){}
    string Member(void){}
};
```