Lab 17
Extra Additives

Objective:

To reinforce skills in defining and manipulating `arrays`, with the `vector` class.

Computers store decimal numbers in binary form to conserve memory. Even though this allows a greater range of values, the largest integer value in many systems is 32,767 and long integers, the largest value is less than 2.2 billion.

Your task is to write a program, using a minimum of three functions that input pairs of numbers between 0 and 100 billion, inclusive. You should then add these numbers together and print the results.

Input:

The input will only consist of digit characters ('0' . . '9'). No commas or other punctuation will be in the data. All values will be non-negative. Since the integers may exceed any binary integer value allowed in the system, we shall read in the values as a string and convert each integer character to its corresponding integer digit and store it in an integer array.

Output:

Your program should print out the results of the addition. Do not print any leading zeros before any answer. Put a loop in your program to run it four times.

Examples:

Input:	Output:
`124235` `7453`	`The sum is: 131688`
`325235` `0`	`The sum is: 325235`
`23523` `100000000000`	`The sum is: 100000023523`
`326346124` `76434124235`	`The sum is: 76760470359`

Use the declarations at right to store your digits.

```
vector<int> largeInt1 (12,0);    // initialize to zero
vector<int> largeInt2 (12,0);    // initialize to zero
vector<int> finalInt (13,0);     // initialize to zero
```

To convert digit characters to digit integers you may use this function.

```
int asciiToInt (char ch)
{
  return (ch - '0');
}
```