

Write a program implementing Gaussian elimination. Run your program with the given input files.

Turn in:

- This problem sheet with your name.
- A summary sheet explaining what you did, how you approached the problem, what was accomplished, what was not accomplished, etc.

Website:

- Create a directory called **3gauss** on your web site and make all your input, output and source files available in this directory. Write the url for the website on this problem sheet.

Input:

The augmented matrix of an $n \times n$ linear system of equations. Each line represents a row of the matrix. The entries of the matrix are separated by spaces.

Output:

The values of the n variables in a row, separated by a space. The output should be the sentence **No unique solution** if the system has no solutions or infinitely many solutions.

Sample input:

```
3 2 1
5 3 2
```

Sample output:

```
1 -1
```

Hints:

- Write separate functions for reading in the data, elimination and back substitution. Start with the given skeleton program.
- Use operator overloading to calculate the scalar multiple of a vector and the sum of two vectors. Take a look at `matrixexample.C`. This will greatly simplify your code.
- You can swap two variables with `swap(var1, var2)`. You need to include the algorithm header file for this.
- It's a good idea to do the elimination step even in the last row. This will find out if the last coefficient is zero without requiring any special code to handle this.
- Use the given makefile.
- It's possible to implement everything in about 75 lines of code.