Trap

Trap is another variant of the "Guess my number" game that has been played on school yards since time began. In this game the player chooses 2 numbers and the one with the secret number responds with whether it is larger, smaller, or between the numbers guessed.

The optimal solution, like with high/low games, is a divide and conquer strategy. However with 2 numbers you should discover quickly that halves are not always the best solution. Played optimally it shouldn't take more than 5 guesses. The game as it's written gives you eight.

Trap is written by Joseph Larson.

TRAP.C	You will need: a C/C++ complier .

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <ctype.h>

#define MAX 100
#define MG 8

int main (void) {
  int x, h, l, try, temp;
  char yesno;

  printf ("Trap\n----\n"
  "In this game you have to try to guess a number between 1 and %d by\n"
  "trapping it. Every guess you type a low and high number seperated by a\n"
  "comma (like \"15, 30\") and you'll be told if the number you are trying to\n"
  "find is between your number. When you think you have it type the same\n"
  "number for both the high and low guess. And remember, you only have %d\n"
  "guesses to find the number\n"
  "Good luck!\n\n", MAX, MG);
  srand (time (NULL));
  do {
    x = rand () % MAX + 1;
    printf ("I have a number. You have %d guesses.\n", MG);
    for (try = 1; try <= MG && !(l == h && h == x); try ++) {
      printf ("\nGuess %d (low , high) : ", try);
      scanf ("%d %*c %d", &l, &h);
      if (l > h) { temp = h; h = l; l = temp;}
      if (l <= x && x <= h)
        printf ("You've trapped my number.\n");
      else printf ("My number is %s than your guesses.\n",
      (l > x) ? "lower" : "higher");
    }
    if (l == h && h == x) printf ("I am undone! You caught my number.\n\n");
    else printf ("Ha ha! That's %d guesses. My number was %d.\n\n", MG, x);
    printf ("Do you want to do play again? (y/n) ");
    while (!isalpha (yesno = getchar ()));
  } while (tolower (yesno) == 'y');
  printf ("Until next time then!\n");
  exit (0);
}
```