# CS 167/467

Final Exam

August 10, 2006

**NAME (please print legibly):** _____

**Your University ID Number:** _____

- **Open book, notes, computer. Take-home. Due Aug 14 at 12:40pm *sharp.***
- **Write directly on the exam. You are welcome to use additional paper or the back of the exam, but if you do so clearly label with the question number.**

| QUESTION | VALUE | SCORE |
|---|---|---|
| 1 | 10 | |
| 2 | 10 | |
| 3 | 10 | |
| 4 | 5 | |
| 5 | 15 | |
| 6 | 10 | |
| 7 | 5 | |
| 8 | 5 | |
| 9 | 5 | |
| 10 | 10 | |
| 11 | 15 | |
| 12 | 15 | |
| TOTAL | 115 | |

**1. (10 points)** Given this snippet:

```
1 char line[6];
2 int counter = 0;
3 while ( fgets(line, 6, stdin) != NULL )
4     counter++;
```

and this input:

```
hello, world
```

What is the value of `counter` after the while loop terminates? (Hint, read the manpage)

**2. (10 points)** Describe a situation where you might use `malloc`. You don't need to give code.

**3. (10 points)** Do you think `structs` are useful? Explain.

**4. (5 points)** Given this snippet:

```
1 int x, y, z;
2 int retval = scanf("%d %d %d", &x, &y, &z);
```

and this input:

```
10 12 foo
```

What is the value of `retval` after line 2 has executed?

Refer to the following code listing for the next few questions:

```
1  #include <stdio.h>
2  #include <ctype.h>
3  #include <stdlib.h>
4  #include <time.h>
5
6  enum {ROCK, PAPER, SCISSORS};
7
8  int main()
9  {
10     /* seed the random number generator */
11     srand(time(0));
12
13     while (1)
14     {
15         char line[1024];
16         int r;
17
18         r = rand() % 3;
19
20         printf("Type rock, paper, or scissors. q to quit.\n");
21         if ( fgets(line, 1024, stdin) == NULL )
22             return 0;
```

```
23
24          switch (tolower(line[0]))
25          {
26          case 'q':
27              return 0;
28
29          case 'r':
30              if (r == SCISSORS)
31                  printf("Scissors. You win!\n\n");
32              else if (r == PAPER)
33                  printf("Paper. I win!\n\n");
34              else
35                  printf("Draw.\n\n");
36              break;
37
38          case 'p':
39              if (r == ROCK)
40                  printf("Rock. You win!\n\n");
41              else if (r == SCISSORS)
42                  printf("Scissors. I win!\n\n");
43              else
44                  printf("Draw.\n\n");
45              break;
46
47          case 's':
48              if (r == PAPER)
49                  printf("Paper. You win!\n\n");
50              else if (r == ROCK)
51                  printf("Rock. I win!\n\n");
52              else
53                  printf("Draw.\n\n");
54              break;
55
56          default:
57              printf("\nI'm sorry, I didn't catch that.\n");
58          }
59      }
```

```
60 }
```

**5. (15 points)** What does this code do?

**6. (10 points)** Under what condition will line 22 be executed?

**7. (5 points)** What happens if you type `Rock`?

**8. (5 points)** What happens if you type `sasquatch`?

**9. (5 points)** Is the computer cheating? Justify your answer.

**10. (10 points)** What would happen if we removed line 36?

**11. (15 points)** Complete the `strlen` function below:

```
int strlen(char *)
{
    char *p = s;
    int counter = 0;









    return counter;
}
```

(If you're not sure what `strlen` does consult the man page or your book.)

**12. (15 points)** *(Extra Credit)* The following code changes `p` to meet some condition. What is that condition?

```
1 int a[SIZE];
2 int *p;
3
4 /* Later, perhaps much later... */
5
6 while (p<a)
7     p += SIZE;
8
9 p = a+((p-a)%SIZE);
```