UNIVERSITY OF MARYLAND
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

ENEE 457
Computer Systems Security
Instructor: Dana Dachman-Soled

Class Exercise–Memory Safety 09/04/19

1 Winning the Lottery

Consider the following simple program:

```
/* lottery.c */

/* This program runs a simple little "lottery" */
/* Your task is to win it with 100% probability */

#include <stdio.h>    /* for printf() */
#include <stdlib.h>   /* for rand() and srand() */
#include <sys/time.h> /* for gettimeofday() */

int your_fcn()
{
    /* Provide three different versions of this, */
    /*  that each win the "lottery" in main().   */
}

int main()
{
    struct timeval tv;    /* Seed the random number generator */
    gettimeofday(&tv, NULL);
    srand(tv.tv_usec);

    const char *sad = ":(";
    const char *happy = ":)";
    int rv;
    rv = your_fcn();

    /* Lottery time */
    if(rv != rand())
        printf("You lose %s\n, sad");
    else
        printf("You win! %s\n", happy);

    return EXIT_SUCCESS;
}
```

This program runs a simple "lottery" by picking a random integer uniformly at random using `rand()`. It draws your number by calling `your_fcn()`, a function that you have complete control over. Your task is to come up with *three* different versions of the function that each win the lottery every time. As a slight hint, note that the only way that we determine whether or not you win is if the program prints ``` ``You win!'' ``` (followed by a newline) at the end. **You are allowed to change anything except the main function.**

You may assume that address space randomization and stack protection is turned *off*.