Dhruv Gupta
Homework 7

Start with readme.c file in a directory.

Compile the file with gcc:

```
gcc readme.c -o readme
```

Execute the newly created exe file:

```
./readme
```

Delete the readme.c file such that the current directory only has the exe file:

```
root@dhruv-MS-7B79:~/Documents/GSU/DataSec/test# ls -la
total 108
drwxr-xr-x 2 dhruv dhruv   4096 Nov 13 17:24 .
drwxr-xr-x 4 dhruv dhruv   4096 Nov 13 17:21 ..
-rwxr-xr-x 1 root  root  100000 Nov 13 17:22 readme
```

Execute the readme file using ./readme:

Prints the code and creates a new .c file

```
root@dhruv-MS-7B79:~/Documents/GSU/DataSec/test# ./readme
/**
@Author Dhruv Gupta

This file is a simple worm that will save the code in memory on the first run and
print the source to command line on the second run.
**/

#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main(){

        //final int static_var_size = 10000;
        const int code_buffer_size = 10000;
        const int exe_buffer_size = 100000; //exe buffer size needs to be >  static_var_size



        static char a[10000] = "hello";
        printf("%s\n", a);
```

Dhruv Gupta
Homework 7

```
// Check if run 1 or run 2
	int flag;
	if(a[0] == 'h') flag=0; else flag=1;

	printf("FLAG ==== > %d\n", flag); //user verification of 1st / 2nd run

if(flag == 0){
	FILE *fp = fopen("readme.c", "r");
	if(!fp){
		printf("Cannot Find .c File ... exiting \n");
		exit(0);
	}

	FILE *fp2 = fopen("readme", "r");

	unsigned char cbuffer [code_buffer_size];
	unsigned char ebuffer [exe_buffer_size];


	int code_len = fread(cbuffer, 1, sizeof(cbuffer), fp);
	int exe_len = fread(ebuffer, 1, sizeof(ebuffer), fp2);

	fclose(fp);
	fclose(fp2);
	//search for string in ebuffer


	int i, j = 0;
		for(i = 0; i < exe_len; i++ )
		{
			// serach for the static variable memory location using its contents as a guide and
replace with .c code
			if(ebuffer[i] == 'h' && ebuffer[i+1] == 'e'    && ebuffer[i+2] == 'l'&&
ebuffer[i+3] == 'l' && ebuffer[i+4] == 'o')
				{
					// ebuffer[i] = 'm';
					for( j=0; j<code_len; j++)
						ebuffer[i++] = cbuffer[j];

				}
		}



		FILE *fp3 = fopen("x.x","w+");
```

Dhruv Gupta
Homework 7

```
                fwrite (ebuffer , sizeof(char), sizeof(ebuffer), fp3);
                fclose(fp3);
                system("mv x.x readme; chmod +x readme");

}else{

        FILE *fp4 = fopen("x.x","w+");
        //fwrite (a , sizeof(char), sizeof(a), fp4);

        fprintf (fp4, "%s", a);
        fclose(fp4);
        system("mv x.x readme.c; chmod 777 readme.c");

}

        return 0;
}
```

FLAG ==== > 1

- Final results readme.c file is back

```
root@dhruv-MS-7B79:~/Documents/GSU/DataSec/test# ls -la
total 112
drwxr-xr-x 2 dhruv dhruv   4096 Nov 13 17:24 .
drwxr-xr-x 4 dhruv dhruv   4096 Nov 13 17:21 ..
-rwxr-xr-x 1 root  root  100000 Nov 13 17:22 readme
-rwxrwxrwx 1 root  root    1755 Nov 13 17:24 readme.c
```

new readme.c file has the same size as the old file.