Seng 360 Assignment 8

[redacted]

Hash Functions

1. Program Design

My C program attempted to use the macro definition header to define what behaviour gets executed. This flag is then used to determine if the good or bad behaviour is executed. My idea is to block comment out the p or q and use those in the executable files. For one file we the starting comment is before the macro definition. This is essentially leading to one file having the flag defined and the other will not. This reaches my goal of trying to have two different behaviours.

2. Steps Taken to Solve this Task

To create these two programs and to make them produce the same hash value first I divided up the file into multiple sections. First, I created a prefix file using the head command and grabbed the first 273 bytes. This includes the macro definition and ends right after that. After that, the MD5 collgen command was used to create two files with the same hash value and different contents. After that I grabbed the remaining bytes of the source program file to use it as a suffix file. After having all these files, I then combined these together using the cat command. This process is shown in the screenshots below. I then added in a block comment statement around the prefix, however, one of the block comments is placed above the ifndef this results in one file only producing the good effect. This still kept the same prefix value as the block comment occurs in both files just changes locations.

```
Terminal
[11/05/20]seed@VM:~$ gcc A9.c
[11/05/20]seed@VM:~$ head -c 273 a.out > prefix
[11/05/20]seed@VM:~$ md5collgen -p prefix -o p q
MD5 collision generator v1.5
by Marc Stevens (http://www.win.tue.nl/hashclash/)

Using output filenames: 'p' and 'q'
Using prefixfile: 'prefix'
Using initial value: 0123456789abcdeffedcba9876543210

Generating first block: ....
Generating second block: S10.....................................................
Running time: 5.47599 s
[11/05/20]seed@VM:~$ tail -c +401 end
[11/05/20]seed@VM:~$ cat prefix >> goodcode
[11/05/20]seed@VM:~$ cat prefix >> badcode
[11/05/20]seed@VM:~$ cat p >> badcode
[11/05/20]seed@VM:~$ cat q >> goodcode
[11/05/20]seed@VM:~$ cat end >> goodcode
[11/05/20]seed@VM:~$ cat end >> badcode
[11/05/20]seed@VM:~$ chmod +x goodcode
[11/05/20]seed@VM:~$ chmod +x badcode
[11/05/20]seed@VM:~$ md5sum goodcode badcode
efd9690e50c582fef867adb2bc438116  goodcode
efd9690e50c582fef867adb2bc438116  badcode
```

3. Observations and Outcome

My program worked as expected. When I ran both my executables, I got the output shown in the screenshot below. What I also observed that when getting the MD5 hash value both c exe files produced the same result. This is also shown in the screenshot below.

```
[11/05/20]seed@VM:~$ chmod +x goodcode
[11/05/20]seed@VM:~$ chmod +x badcode
[11/05/20]seed@VM:~$ md5sum goodcode badcode
efd9690e50c582fef867adb2bc438116  goodcode
efd9690e50c582fef867adb2bc438116  badcode
[11/05/20]seed@VM:~$ ./ goodcode
:) Good program
[11/05/20]seed@VM:~$ ./ badcode
:(( Bad program :((
```