Lab 2 Report

Nicholas Jurden 2415098

1. The problem in the original code was a lack of "savestring" method acting on space that was recently freed. Because the space was freed in the initial loop, the program was trying to free a pointer that was no longer referencing allocated space in memory.
2. I struggled to diagnose the problem initially on my Mac, however, upon switching to the in lab linux machines, the terminal responses to lab commands gave the exact line of the error. After determining where the error was occurring, I searched the code for similar patterns to the code on that line and found an instance where savestring was used.
3. My solution fixes the problem, as shown in the savestring documentation, by allocating new memory and calling strcpy on the string that is passed in, then returning the new memory location populated with the string. Documentation is as follows (found on http://bashcookbook.com/bashinfo/source/readline-6.3/savestring.c):

```
char *
savestring (s)
     const char *s;
{
  char *ret;

  ret = (char *)xmalloc (strlen (s) + 1);
  strcpy (ret, s);
  return ret;
}
```