Problem 1 of 8

I/O Operations

This question will test your understanding of how file I/O works. Assume that all system calls succeed.

The file `foobar.txt` contains the string `eucalyptus treat` .

```
#include <stdio.h>
#include <fcntl.h>
#include <unistd.h>

int main()
{
  char buf[7] = {0}; // This initializes an array of all 0s
  int fd1 = open("foobar.txt", O_RDWR);
  int fd2 = open("foobar.txt", O_RDWR);
  int fd3 = 0;

  read(fd1, buf, 2);

  dup2(fd1, fd3);

  write(fd3, "kazillion!", 10);
  write(fd2, "a-", 2); // that's a dash !

  read(fd1, &buf[2], 2);
  read(fd2, &buf[4], 2);

  printf("%s\n", buf);

  return 0;
}
```

1) What gets printed by the program?

Output:

Now consider the following program, which will test your knowledge of file I/O as it applies to processes. Use the same assumptions as in the previous question, including the text contained in the file.

```
#include <stdio.h>
#include <fcntl.h>
#include <unistd.h>
#include <sys/wait.h>

int main()
{
  char buf[7] = {0};
  int fd1 = open("foobar.txt", O_RDWR);
  int fd2 = open("foobar.txt", O_RDWR);

  if (fork() == 0)
  {
    dup2(fd1, fd2);
    read(fd2, &buf, 3);
    printf("%s", buf);
  }
  else
  {
    waitpid(-1, NULL, 0);
    read(fd2, &buf, 2);
    read(fd1, &buf[2], 2);
    printf("%s\n", buf);
  }

  return 0;
}
```

2) What gets printed by the program?

Output: