Student ID: ______________________________

This sheet will not be graded (feel free to write on it), but you must turn it in at the end of the exam. Please do not flip this sheet over until the start of the exam.

Function Definitions

```
char *fgets(char *s, int size, FILE *stream);
```

fgets() reads in at most one less than size characters from stream and stores them into the buffer pointed to by s. Reading stops after an EOF or a newline. If a newline is read, it is stored into the buffer. A terminating null byte (\0) is stored after the last character in the buffer.

```
size_t fread(void *ptr, size_t size, size_t nmemb, FILE *stream);
```

The function fread() reads nmemb items of data, each size bytes long, from the stream pointed to by stream, storing them at the location given by ptr.

```
char *gets(char *s);
```

gets() reads a line from stdin into the buffer pointed to by s until either a terminating newline or EOF, which it replaces with a null byte (\0).

```
int printf(const char *format, ...);
```

printf() produces output according to the format string format.

This is a copy of the code snippet from Question 3 (The Big Reveal).

```
void reveal(ch r* format, ch r* identity) {
    printf( Drumroll please...\n );
    printf(format, identity);
    printf( *shocked pikachu face*\n );
}

void identify(ch r* format) {
    ch r identity[16];
    gets(identity);
    reveal(format, identity);
}

int main() {
    identify( EvanBot is %s\n );
    return 0;
}
```

This is a copy of the code snippet from Question 4 (Here we go again...).

```
struct lockbox {
    ch r name[4];
    ch r *invitation_ptr;
    ch r *file_ptr;
};

void func() {
    lockbox *alicebob = malloc(sizeof(lockbox));
    alicebob->invitation_ptr = (ch r*) malloc(108);
    alicebob->file_ptr = invitation_ptr;

    fread(alicebob->name, 5, 1, stdin);
    fread(alicebob->invitation_ptr, 3, 1, stdin);
    fread(alicebob->file_ptr, 108, 1, stdin);
}
```