```c
#include <stdio.h>
#include <stdlib.h>

int main() {
    char *text;
    text = (char*)malloc(sizeof(char) * 10);

    int i;
    for (i = 0; i < 6; ++i)
        text[i] = "catdog"[i];
    text[i] = '\0';

    const char *the_text = text;
    printf("text is %s\n", text);
    printf("the_text is %s\n", the_text);
    printf("addr of text is %p\n", text);

    text = (char*)realloc(text, sizeof(char) * 10000);
    printf("text is %s\n", text);
    printf("addr of text is %p\n", text);

    for (i = 0; i < 6; ++i)
        printf("the_text is %d\n", the_text[i]);

    struct book {
        char *text;
    };
    struct book *bookp = (struct book*)malloc(sizeof(struct
     book));
    if (bookp == NULL) {
        fprintf(stderr, "cannot allocate %zu bytes\n",
         sizeof(struct book));
        return 1;
    }
    bookp->text = text;
    printf("book text is: %s\n", bookp->text);
    free(bookp);
    bookp = NULL;
```

```
    // purposely entered garbage there
    unsigned char *ptr = (unsigned char*)bookp;
    for (i = 0; i < sizeof(struct book); ++i)
        ptr[i] = i+1;

    struct book *bookp2 = (struct book*)malloc(sizeof(struct
     book));
    printf("book text is: %s\n", bookp->text);

    free(text);
    return 0;
}
```