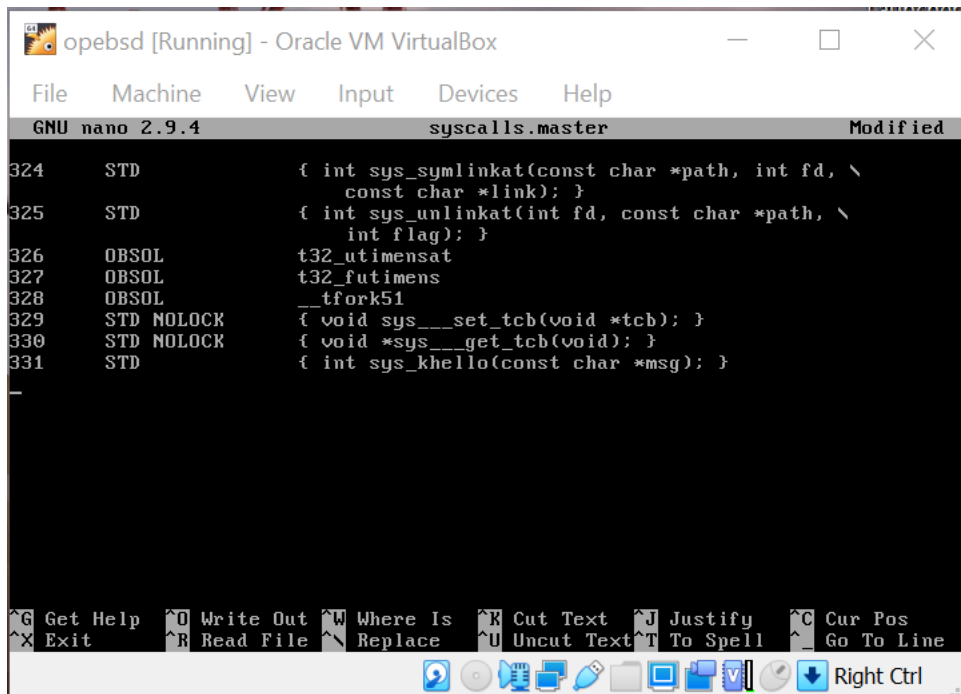


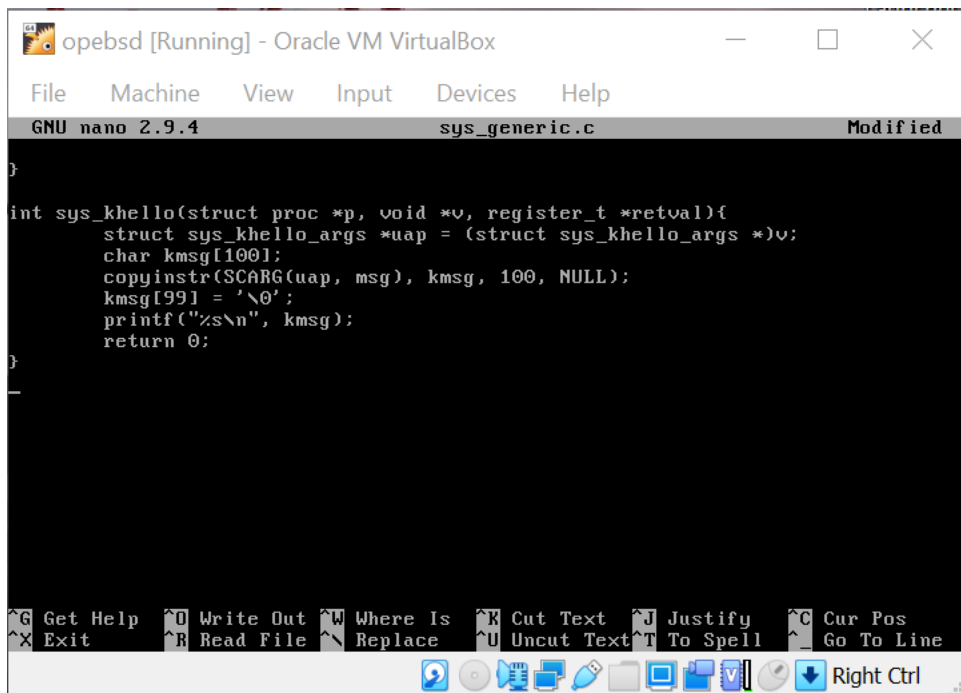
Exercitiul 2



The screenshot shows a window titled "opebsd [Running] - Oracle VM VirtualBox". Inside, the GNU nano 2.9.4 text editor is open, editing the file "syscalls.master". The editor displays a list of system calls with their line numbers (324 to 331) and their definitions. The definitions are as follows:

```
324  STD      { int sys_symlinkat(const char *path, int fd, \
325  STD      { int sys_unlinkat(int fd, const char *path, \
326  OBSOLETE t32_utimensat
327  OBSOLETE t32_futimens
328  OBSOLETE __tfork51
329  STD NOLOCK { void sys___set_tcb(void *tcb); }
330  STD NOLOCK { void *sys___get_tcb(void); }
331  STD      { int sys_khello(const char *msg); }
```

The bottom of the window shows a toolbar with various icons and a status bar with the text "Right Ctrl".



The screenshot shows a window titled "opebsd [Running] - Oracle VM VirtualBox". Inside, the GNU nano 2.9.4 text editor is open, editing the file "sys_generic.c". The editor displays the definition of the "sys_khello" function. The code is as follows:

```
int sys_khello(struct proc *p, void *v, register_t *retval){
    struct sys_khello_args *uap = (struct sys_khello_args *)v;
    char kmsg[100];
    copyinstr(SCARG(uap, msg), kmsg, 100, NULL);
    kmsg[99] = '\0';
    printf("%s\n", kmsg);
    return 0;
}
```

The bottom of the window shows a toolbar with various icons and a status bar with the text "Right Ctrl".

opebsd [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

GNU nano 2.9.4 hello.c

```
int main(){
    syscall(331, "Buna seara Bucuresti");
    return 0;
}
```

[Read 4 lines]

Get Help Write Out Where Is Cut Text Justify Cur Pos
Exit Read File Replace Uncut Text To Spell Go To Line

Right Ctrl

opebsd [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
syscall(331, "Buna seara Bucuresti");
return 0;
}
```

opebsd# gcc hello.c -o test

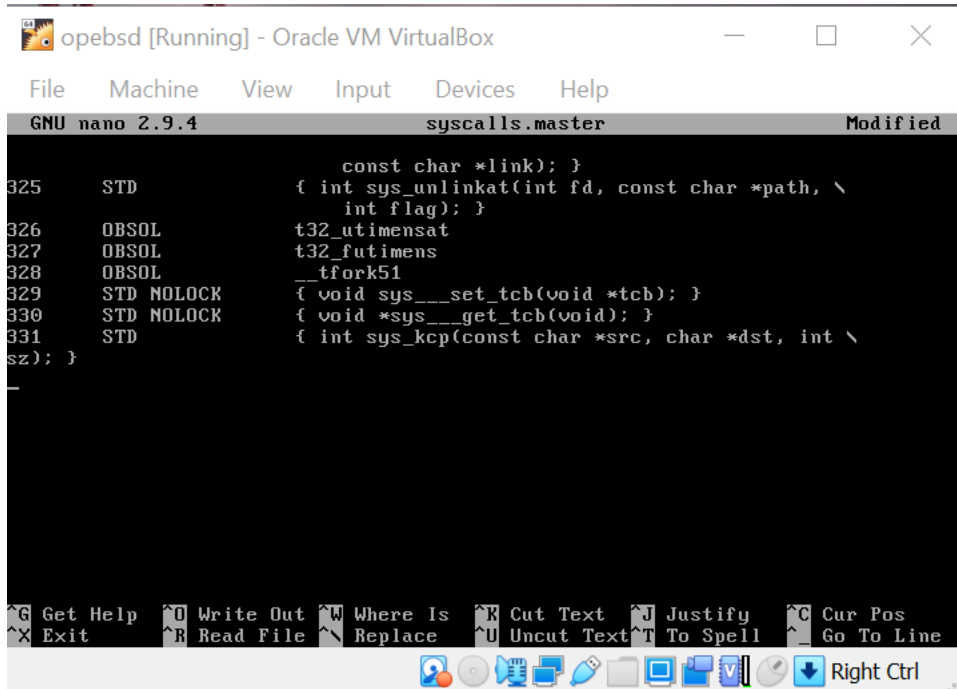
opebsd# ./test

Buna seara Bucuresti

opebsd#

Right Ctrl

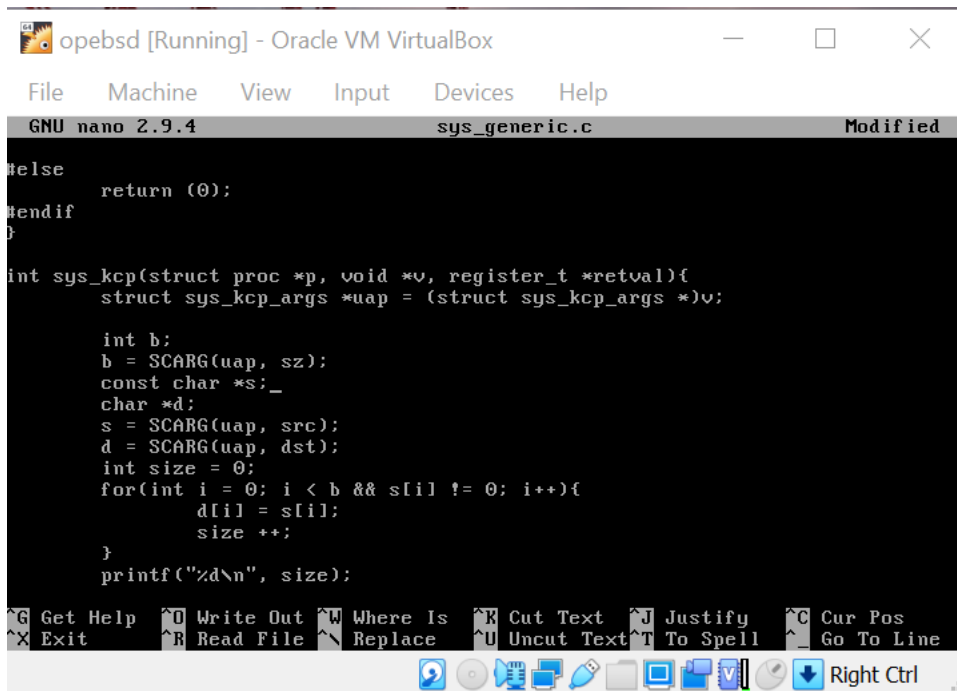
Exercitiul 3



The screenshot shows a nano editor window titled "opebsd [Running] - Oracle VM VirtualBox". The menu bar includes File, Machine, View, Input, Devices, and Help. The status bar at the top indicates "GNU nano 2.9.4", "syscalls.master", and "Modified". The editor displays the following code:

```
const char *link); }
{ int sys_unlinkat(int fd, const char *path, \
  int flag); }
325   STD      t32_utimensat
326   OBSOL   t32_futimens
327   OBSOL   __tfork51
328   OBSOL   { void sys___set_tcb(void *tcb); }
329   STD NOLOCK { void *sys___get_tcb(void); }
330   STD NOLOCK { int sys_kcp(const char *src, char *dst, int \
331   STD      sz); }
-
```

The bottom status bar shows various nano editor shortcuts: ^G Get Help, ^X Exit, ^O Write Out, ^R Read File, ^W Where Is, ^M Replace, ^K Cut Text, ^U Uncut Text, ^J Justify, ^T To Spell, ^C Cur Pos, and ^_ Go To Line. A toolbar with icons for file operations and a "Right Ctrl" button is also visible.



The screenshot shows a nano editor window titled "opebsd [Running] - Oracle VM VirtualBox". The menu bar includes File, Machine, View, Input, Devices, and Help. The status bar at the top indicates "GNU nano 2.9.4", "sys_generic.c", and "Modified". The editor displays the following code:

```
#else
    return (0);
#endif
}

int sys_kcp(struct proc *p, void *v, register_t *retval){
    struct sys_kcp_args *uap = (struct sys_kcp_args *)v;

    int b;
    b = SCARG(uap, sz);
    const char *s;
    char *d;
    s = SCARG(uap, src);
    d = SCARG(uap, dst);
    int size = 0;
    for(int i = 0; i < b && s[i] != 0; i++){
        d[i] = s[i];
        size ++;
    }
    printf("%d\n", size);
}
```

The bottom status bar shows various nano editor shortcuts: ^G Get Help, ^X Exit, ^O Write Out, ^R Read File, ^W Where Is, ^M Replace, ^K Cut Text, ^U Uncut Text, ^J Justify, ^T To Spell, ^C Cur Pos, and ^_ Go To Line. A toolbar with icons for file operations and a "Right Ctrl" button is also visible.

oopenbsd [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

GNU nano 2.9.4 cp.c Modified

```
#include <stdio.h>
#include <string.h>
int main(){
    const char s[100] = "Buna seara!!!";
    char d[100];
    strcpy(d, "..... Bucuresti");
    syscall(331, s, d, 10);
    printf("%s\n", d);
    return 0;
}
```

Get Help Write Out Where Is Cut Text Justify Cur Pos
Exit Read File Replace Uncut Text To Spell Go To Line

Right Ctrl

oopenbsd [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
strcpy(d, "..... Bucuresti");
syscall(331, s, d, 10);
printf("%s\n", d);
return 0;
}
```

openbsd# gcc cp.c -o test
/tmp/ccDcqbX1.o: In function 'main':
cp.c:(.text+0x59): warning: strcpy() is almost always misused, please use strncpy()
openbsd# ./test
10
Buna seara Bucuresti
openbsd#

Right Ctrl