

CS2006 Operating Systems



Project Report

Ice Cream Factory System Call

Course Instructors:

Dr Ghufraan Ahmed

Miss Anaum Hamid

Group Members

21K-3161 Muhammad Ahmed

21K-4556 Muhammad Anas

21K-3323 Muhammad Shaheer

Introduction:

The ice cream industry is a multi-billion dollar industry that continues to grow every year. With the increasing demand for ice cream, there is a need for efficient management of ice cream production processes. This project aims to implement a system call that will facilitate the management of ice cream flavors and their associated recipes in an operating system.

Objective:

The main objective of this project is to implement a system call in an operating system that simulates an ice cream factory system. The system call should allow users to enter the ice cream shop, wait for their turn, get their specified flavors, have toppings on their ice cream, get billed and then leave the shop. Since there can be several customers our main objective is to provide process Synchronization using Semaphores to prevent collisions of different process of Customers

The problem is how to design a discipline of behavior (a concurrent algorithm) such that no customer will be left behind, i.e., each can continue to have the ice cream of their favorite flavor with toppings.

This project objectives would be:

- Create a System Call
- Use the System Call in the Project.
- Thread Synchronization
- Semaphores
- Deadlock Handling

Background:

The reason for creating an ice cream factory system call is to facilitate the management of ice cream production processes within an operating system. The ice cream industry is a complex and rapidly growing industry, and managing the various stages of the ice cream production process can be challenging.

By implementing an ice cream factory system call, users can handle operations there through the operating system. This system call provides a standardized and efficient way of managing ice cream flavors and toppings, reducing the potential for errors and ensuring consistency across the production process.

Moreover, the ice cream factory system call can be integrated with other system-level programming tools, such as file systems, to provide a complete solution for managing the production process at advanced level. This integration can lead to increased efficiency, improved quality control, and reduced costs.

Overall, the ice cream factory system call provides a valuable tool for managing the ice cream production process within an operating system, streamlining the production process, and improving overall efficiency. So this is the main reason to implement the system call which is the background that why we use this through system call.

Platforms:

The platform we used was Linux Operating System using Oracle Virtual Box.

Version of Ubuntu used is 16.04 LTS.

Languages:

The language used for this project is C Language. C language is often used for system call programming because it provides low-level access to system resources and hardware. C is a compiled language, meaning that the code is translated directly into machine code, making it faster and more efficient than interpreted languages.

Methodology:

- Components of ice creams are made by Thread at each counter.
- These Threads will be executing concurrently with other threads.
- Then it will write data on a thread by allocating flavors to our ice cream and then it will call another processor.
- Local semaphore will be used there. It will also show as Ice cream or thread has been successfully created. Semaphores used in our code are that of tickets, 3 types of flavors, 2 types of topping and payment.
- Order of ice cream will arrive at first counters then a sequence of counters will accomplish the order by providing different flavors for their ice-creams.
- They will be served in multiple flavors including in the critical section and after that, it will wait for 5 sec so that ice cream flavors will be assigned to ice cream without any hurdle. Toppings would be assigned after flavors.
- After completely filling, Ice cream orders will be ready to distribute, preventing deadlock. Then the customer can leave the shop

Kernel Configuration Images:

To implement the system call we need to update the version of kernel which is the elementary step leading towards a system call. This configuration involves making main.c file which invokes system call at kernel level along with alterations in config files such as creating a Make file, adding system call to a new index at the system call table, adding function's prototype in the syscall.h file and then adding the system call name in the Original Make file.

We used the following link to add the system call: <https://youtu.be/qcsKGWshRZE>.

Attached below are the Screenshots after Kernel Configuration:

Sys call code at kernel level:

```
PS.c (~/.Downloads/linux-4.19.282/PS) - gedit
Open [v] [f]

#include <linux/kernel.h>
#include <linux/syscalls.h>

SYSCALL_DEFINE2(PS, int, a, char*, buff)
{
    printk("\n[ICECREAM SHOP]: Customer[%d]:%s",a,buff);
    return 0;
}
```

Sys_call64.tbl:

```
syscall_64.tbl (~/.Downloads/linux-4.19.282/arch/x86/entry/syscalls) - gedit
Open [v] [f]

327 64 preadv2 __x64_sys_preadv2
328 64 pwritev2 __x64_sys_pwritev2
329 common pkey_mprotect __x64_sys_pkey_mprotect
330 common pkey_alloc __x64_sys_pkey_alloc
331 common pkey_free __x64_sys_pkey_free
332 common statx __x64_sys_statx
333 common io_pgetevents __x64_sys_io_pgetevents
334 common rseq __x64_sys_rseq
335 common PS __x64_sys_PS
#
# x32-specific system call numbers start at 512 to avoid cache impact
# for native 64-bit operation. The __x32_compat_sys stubs are created
# on-the-fly for compat_sys_*( ) compatibility system calls if X86_X32
# is defined.
#
```

Syscall.h file:

```
syscalls.h (~/.Downloads/linux-4.19.282/include/linux) - gedit
Open [v] [f]

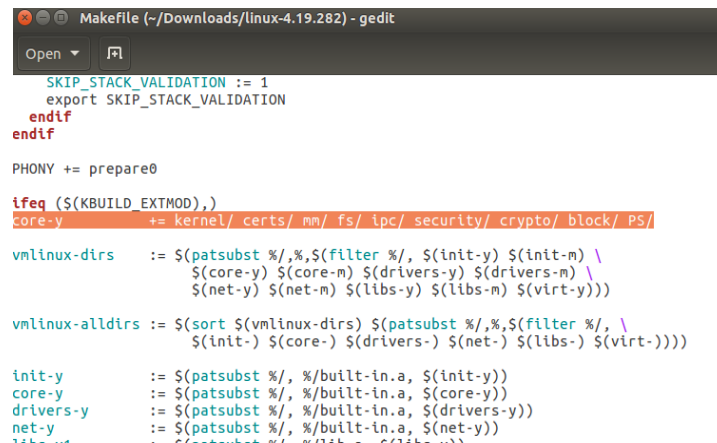
*/__close_fd().
static inline int ksys_close(unsigned int fd)
{
    return __close_fd(current->files, fd);
}

extern long do_sys_open(int dfd, const char __user *filename, int flags,
umode_t mode);
static inline long ksys_open(const char __user *filename, int flags,
umode_t mode)
{
    if (force_o_largefile())
        flags |= O_LARGEFILE;
    return do_sys_open(AT_FDCWD, filename, flags, mode);
}

extern long do_sys_truncate(const char __user *pathname, loff_t length);
static inline long ksys_truncate(const char __user *pathname, loff_t length)
{
    return do_sys_truncate(pathname, length);
}

static inline unsigned int ksys_personality(unsigned int personality)
{
    unsigned int old = current->personality;
    if (personality != 0xffffffff)
        set_personality(personality);
    return old;
}
asmlinkage long sys_PS(int, char*);
#endif
```

Kernel's Make file:



```
Makefile (~/Downloads/linux-4.19.282) - gedit
Open  [icon]

SKIP_STACK_VALIDATION := 1
export SKIP_STACK_VALIDATION
endif
endif

PHONY += prepare0

ifeq ($(KBUILD_EXTMOD),)
core-y += kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ block/ PS/

vmlinux-dirs := $(patsubst %/,%, $(filter %/, $(init-y) $(init-m) \
$(core-y) $(core-m) $(drivers-y) $(drivers-m) \
$(net-y) $(net-m) $(libs-y) $(libs-m) $(virt-y)))

vmlinux-alldirs := $(sort $(vmlinux-dirs) $(patsubst %/,%, $(filter %/, \
$(init-) $(core-) $(drivers-) $(net-) $(libs-) $(virt-))))

init-y := $(patsubst %/, %/built-in.a, $(init-y))
core-y := $(patsubst %/, %/built-in.a, $(core-y))
drivers-y := $(patsubst %/, %/built-in.a, $(drivers-y))
net-y := $(patsubst %/, %/built-in.a, $(net-y))
libs-y := $(patsubst %/, %/built-in.a, $(libs-y))
virt-y := $(patsubst %/, %/built-in.a, $(virt-y))
```

Make File



```
Open  [icon]

obj-y := PS.o
```

Results:

Attached below are the outputs received upon compiling the code. This code is implemented at user level, and it calls system call at index 335 recursively whenever it is needed. In user level code we first ask the number of customers which can be within a range of 1-30 at a time to manage all resources effectively. At the end the Business Journal will help us find the total Revenue generated with the number of customers entertained at the shop.

Ice Cream Factory Project

```
k214556@k214556-VirtualBox:~$ ./t

Enter Number Of Customers [1-30]: 20

Business Journal - At Closing

Number Of Customers: 20
Revenue Generated: $ 117.060000
Tickets Remaining: 10
```

```
[ 2779.659195] [ICECREAM SHOP]: Customer[106]:Got Flavour[0].
[ 2780.678171] [ICECREAM SHOP]: Customer[101]:Leaving Ice-Cream Shop.

[ 2780.678178] [ICECREAM SHOP]: Customer[112]:Billed: $ 6.020000.
[ 2780.678376] [ICECREAM SHOP]: Customer[107]:Got Flavour(s). Leaving Flavor Counter
[ 2780.678395] [ICECREAM SHOP]: Customer[107]:Leaving Topping Counter.
[ 2780.678505] [ICECREAM SHOP]: Customer[104]:Got Flavour[2].
[ 2780.689375] [ICECREAM SHOP]: Customer[106]:Got Flavour[1].
[ 2780.689499] [ICECREAM SHOP]: Customer[100]:Got Flavour[0].
[ 2781.657764] [ICECREAM SHOP]: Customer[113]:Leaving Ice-Cream Shop.

[ 2781.662176] [ICECREAM SHOP]: Customer[119]:Billed: $ 6.020000.
[ 2781.680778] [ICECREAM SHOP]: Customer[104]:Got Flavour(s). Leaving Flavor Counter
[ 2781.680804] [ICECREAM SHOP]: Customer[104]:Leaving Topping Counter.
[ 2781.689520] [ICECREAM SHOP]: Customer[106]:Got Flavour[2].
[ 2781.689733] [ICECREAM SHOP]: Customer[100]:Got Flavour[1].
[ 2781.689773] [ICECREAM SHOP]: Customer[102]:Got Flavour[0].
[ 2782.680079] [ICECREAM SHOP]: Customer[112]:Leaving Ice-Cream Shop.

[ 2782.680213] [ICECREAM SHOP]: Customer[107]:Billed: $ 6.020000.
[ 2782.690098] [ICECREAM SHOP]: Customer[106]:Got Flavour(s). Leaving Flavor Counter
[ 2782.690107] [ICECREAM SHOP]: Customer[102]:Got Flavour[1].
[ 2782.690110] [ICECREAM SHOP]: Customer[106]:Leaving Topping Counter.
[ 2782.690119] [ICECREAM SHOP]: Customer[100]:Got Flavour[2].
[ 2782.690120] [ICECREAM SHOP]: Customer[109]:Got Flavour[0].
[ 2783.662658] [ICECREAM SHOP]: Customer[119]:Leaving Ice-Cream Shop.
```

```
k214556@k214556-VirtualBox:~$ sudo dmesg -c
[ 2771.610505] [ICECREAM SHOP]: Customer[101]:Got Ticket.
[ 2771.610521] [ICECREAM SHOP]: Customer[102]:Got Ticket.
[ 2771.610559] [ICECREAM SHOP]: Customer[100]:Got Ticket.
[ 2771.610716] [ICECREAM SHOP]: Customer[105]:Got Ticket.
[ 2771.610736] [ICECREAM SHOP]: Customer[103]:Got Ticket.
[ 2771.611311] [ICECREAM SHOP]: Customer[104]:Got Ticket.
[ 2771.612670] [ICECREAM SHOP]: Customer[108]:Got Ticket.
[ 2771.612740] [ICECREAM SHOP]: Customer[107]:Got Ticket.
[ 2771.612766] [ICECREAM SHOP]: Customer[106]:Got Ticket.
[ 2771.612854] [ICECREAM SHOP]: Customer[112]:Got Ticket.
[ 2771.612867] [ICECREAM SHOP]: Customer[113]:Got Ticket.
[ 2771.613125] [ICECREAM SHOP]: Customer[118]:Got Ticket.
[ 2771.613141] [ICECREAM SHOP]: Customer[117]:Got Ticket.
[ 2771.613153] [ICECREAM SHOP]: Customer[111]:Got Ticket.
[ 2771.613163] [ICECREAM SHOP]: Customer[114]:Got Ticket.
[ 2771.613171] [ICECREAM SHOP]: Customer[110]:Got Ticket.
[ 2771.613195] [ICECREAM SHOP]: Customer[115]:Got Ticket.
[ 2771.613211] [ICECREAM SHOP]: Customer[109]:Got Ticket.
[ 2771.613253] [ICECREAM SHOP]: Customer[116]:Got Ticket.
[ 2771.613314] [ICECREAM SHOP]: Customer[119]:Got Ticket.
[ 2773.610756] [ICECREAM SHOP]: Customer[101]:Got Flavour[0].
[ 2774.647015] [ICECREAM SHOP]: Customer[101]:Got Flavour[1].
[ 2774.647133] [ICECREAM SHOP]: Customer[113]:Got Flavour[0].
[ 2775.647133] [ICECREAM SHOP]: Customer[101]:Got Flavour[2].
[ 2775.647587]
```

```
[ 2792.757491] [ICECREAM SHOP]: Customer[108]:Leaving Topping Counter.
[ 2792.757610] [ICECREAM SHOP]: Customer[111]:Got Flavour[2].
[ 2793.755396] [ICECREAM SHOP]: Customer[110]:Got Flavour[0].
[ 2793.760033] [ICECREAM SHOP]: Customer[118]:Leaving Ice-Cream Shop.
[ 2793.760105] [ICECREAM SHOP]: Customer[110]:Got Flavour[1].
[ 2793.760281] [ICECREAM SHOP]: Customer[115]:Billed: $ 6.020000.
[ 2793.760283] [ICECREAM SHOP]: Customer[111]:Got Flavour(s). Leaving Flavor Counter
[ 2793.760336] [ICECREAM SHOP]: Customer[111]:Leaving Topping Counter.
[ 2793.760340] [ICECREAM SHOP]: Customer[114]:Got Flavour(s). Leaving Flavor Counter
[ 2794.755690] [ICECREAM SHOP]: Customer[114]:Leaving Topping Counter.
[ 2794.757462] [ICECREAM SHOP]: Customer[116]:Leaving Ice-Cream Shop.
[ 2794.760206] [ICECREAM SHOP]: Customer[108]:Billed: $ 6.020000.
[ 2794.760211] [ICECREAM SHOP]: Customer[110]:Got Flavour(s). Leaving Flavor Counter
[ 2795.760662] [ICECREAM SHOP]: Customer[110]:Leaving Topping Counter.
[ 2795.760765] [ICECREAM SHOP]: Customer[111]:Billed: $ 6.020000.
[ 2795.785179] [ICECREAM SHOP]: Customer[114]:Billed: $ 4.350000.
[ 2796.757731] [ICECREAM SHOP]: Customer[115]:Leaving Ice-Cream Shop.
[ 2796.760626] [ICECREAM SHOP]: Customer[108]:Leaving Ice-Cream Shop.
[ 2797.768789] [ICECREAM SHOP]: Customer[110]:Billed: $ 4.350000.
[ 2797.768823] [ICECREAM SHOP]: Customer[114]:Leaving Ice-Cream Shop.
[ 2798.760740] [ICECREAM SHOP]: Customer[111]:Leaving Ice-Cream Shop.
[ 2798.760740] [ICECREAM SHOP]: Customer[110]:Leaving Ice-Cream Shop.
k214556@k214556-VirtualBox:~$ ./t
```


Ice Cream Factory Project

```
[ 2783.690268] [ICECREAM SHOP]: Customer[100]:Got Flavour(s). Leaving Flavor Counter
[ 2783.690288] [ICECREAM SHOP]: Customer[100]:Leaving Topping Counter.
[ 2783.690323] [ICECREAM SHOP]: Customer[102]:Got Flavour[2].
[ 2783.690491] [ICECREAM SHOP]: Customer[109]:Got Flavour[1].
[ 2783.690549] [ICECREAM SHOP]: Customer[103]:Got Flavour[0].
[ 2783.691826] [ICECREAM SHOP]: Customer[104]:Billed: $ 6.020000.
[ 2784.681631] [ICECREAM SHOP]: Customer[107]:Leaving Ice-Cream Shop.
[ 2784.690704] [ICECREAM SHOP]: Customer[106]:Billed: $ 6.020000.
[ 2784.690777] [ICECREAM SHOP]: Customer[102]:Got Flavour(s). Leaving Flavor Counter
[ 2784.690795] [ICECREAM SHOP]: Customer[102]:Leaving Topping Counter.
[ 2784.690819] [ICECREAM SHOP]: Customer[109]:Got Flavour[2].
[ 2784.690820] [ICECREAM SHOP]: Customer[103]:Got Flavour[1].
[ 2784.690867] [ICECREAM SHOP]: Customer[105]:Got Flavour[0].
[ 2785.690893] [ICECREAM SHOP]: Customer[109]:Got Flavour(s). Leaving Flavor Counter
[ 2785.690916] [ICECREAM SHOP]: Customer[109]:Leaving Topping Counter.
[ 2785.701481] [ICECREAM SHOP]: Customer[103]:Got Flavour[2].
[ 2785.701599] [ICECREAM SHOP]: Customer[104]:Leaving Ice-Cream Shop.
[ 2785.701636] [ICECREAM SHOP]: Customer[100]:Billed: $ 6.020000.
[ 2785.716559] [ICECREAM SHOP]: Customer[105]:Got Flavour[1].
[ 2785.716672] [ICECREAM SHOP]: Customer[117]:Got Flavour[0].
[ 2786.697577] [ICECREAM SHOP]: Customer[106]:Leaving Ice-Cream Shop.
[ 2786.719989] [ICECREAM SHOP]: Customer[117]:Got Flavour[1].
```

```
[ 2775.647587] [ICECREAM SHOP]: Customer[113]:Got Flavour[1].
[ 2775.647665] [ICECREAM SHOP]: Customer[112]:Got Flavour[0].
[ 2776.647318] [ICECREAM SHOP]: Customer[101]:Got Flavour(s). Leaving Flavor Counter
[ 2776.647348] [ICECREAM SHOP]: Customer[101]:Leaving Topping Counter.
[ 2776.656912] [ICECREAM SHOP]: Customer[113]:Got Flavour[2].
[ 2776.657000] [ICECREAM SHOP]: Customer[112]:Got Flavour[1].
[ 2776.657152] [ICECREAM SHOP]: Customer[119]:Got Flavour[0].
[ 2777.657163] [ICECREAM SHOP]: Customer[113]:Got Flavour(s). Leaving Flavor Counter
[ 2777.657187] [ICECREAM SHOP]: Customer[113]:Leaving Topping Counter.
[ 2777.657376] [ICECREAM SHOP]: Customer[112]:Got Flavour[2].
[ 2777.657383] [ICECREAM SHOP]: Customer[119]:Got Flavour[1].
[ 2777.657395] [ICECREAM SHOP]: Customer[107]:Got Flavour[0].
[ 2778.647625] [ICECREAM SHOP]: Customer[101]:Billed: $ 6.020000.
[ 2778.657567] [ICECREAM SHOP]: Customer[112]:Got Flavour(s). Leaving Flavor Counter
[ 2778.657588] [ICECREAM SHOP]: Customer[112]:Leaving Topping Counter.
[ 2778.657751] [ICECREAM SHOP]: Customer[104]:Got Flavour[0].
[ 2778.657953] [ICECREAM SHOP]: Customer[119]:Got Flavour[2].
[ 2778.658006] [ICECREAM SHOP]: Customer[107]:Got Flavour[1].
[ 2779.657545] [ICECREAM SHOP]: Customer[113]:Billed: $ 6.020000.
[ 2779.658569] [ICECREAM SHOP]: Customer[119]:Got Flavour(s). Leaving Flavor Counter
[ 2779.658589] [ICECREAM SHOP]: Customer[119]:Leaving Topping Counter.
[ 2779.658696] [ICECREAM SHOP]: Customer[104]:Got Flavour[1].
[ 2779.658795] [ICECREAM SHOP]: Customer[107]:Got Flavour[2].
[ 2779.659195]
```

```
[ 2786.741835] [ICECREAM SHOP]: Customer[102]:Billed: $ 6.020000.
[ 2786.752581] [ICECREAM SHOP]: Customer[103]:Got Flavour(s). Leaving Flavor Counter
[ 2786.752651] [ICECREAM SHOP]: Customer[103]:Leaving Topping Counter.
[ 2786.752679] [ICECREAM SHOP]: Customer[105]:Got Flavour[2].
[ 2786.752689] [ICECREAM SHOP]: Customer[118]:Got Flavour[0].
[ 2787.694954] [ICECREAM SHOP]: Customer[109]:Billed: $ 6.020000.
[ 2787.704918] [ICECREAM SHOP]: Customer[100]:Leaving Ice-Cream Shop.
[ 2787.753012] [ICECREAM SHOP]: Customer[105]:Got Flavour(s). Leaving Flavor Counter
[ 2787.753029] [ICECREAM SHOP]: Customer[105]:Leaving Topping Counter.
[ 2787.753111] [ICECREAM SHOP]: Customer[117]:Got Flavour[2].
[ 2787.753505] [ICECREAM SHOP]: Customer[118]:Got Flavour[1].
[ 2787.753548] [ICECREAM SHOP]: Customer[116]:Got Flavour[0].
[ 2788.754564] [ICECREAM SHOP]: Customer[117]:Got Flavour(s). Leaving Flavor Counter
[ 2788.754588] [ICECREAM SHOP]: Customer[117]:Leaving Topping Counter.
[ 2788.754780] [ICECREAM SHOP]: Customer[115]:Got Flavour[0].
[ 2788.754853] [ICECREAM SHOP]: Customer[118]:Got Flavour[2].
[ 2788.754932] [ICECREAM SHOP]: Customer[116]:Got Flavour[1].
[ 2788.768542] [ICECREAM SHOP]: Customer[103]:Billed: $ 6.020000.
[ 2788.776241] [ICECREAM SHOP]: Customer[102]:Leaving Ice-Cream Shop.
[ 2789.695033] [ICECREAM SHOP]: Customer[109]:Leaving Ice-Cream Shop.
[ 2789.753861] [ICECREAM SHOP]: Customer[105]:Billed: $ 6.020000.
[ 2789.754970] [ICECREAM SHOP]: Customer[118]:Got Flavour(s). Leaving Flavor Counter
[ 2789.754991]
```

Conclusion:

To ensure all the customers, have ice cream, firstly tickets should be greater than customers, if tickets less than customers then deadlock occurs, so more customers won't be entertained. Race Condition will occur here a thread will proceed even though flavors have been finished. If this is not used, then a thread will check each condition below which will be time consuming. If any of the threads did not get any flavor, the Race condition remains 0 and threads will exit.

In conclusion, this project aims to implement a system call in an operating system that will facilitate the management of ice cream flavors and their associated recipes. The project will be implemented in the Linux operating system using the C programming language and the file system. The project will involve the design, implementation, testing, and integration of the system call with the file system.

