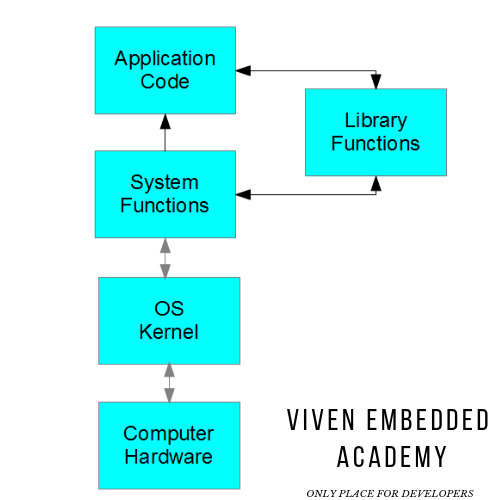
**CMPE 312 Operating Systems Assignment**

**System Calls**

**1-) What is a system call? Please briefly explain the basics of system call, why we need them, and what we can achieve with them, how they are implemented etc.**

System call is the services provided by Linux kernel. In C programming, it often uses functions defined in “libc” which provides a wrapper for many system calls. Manual page section 2 provides more information about system calls. To get an overview, use “man 2 intro” in a command shell. Also, it is controlled entry point into the kernel. System calls serves for



* It allows the process to send request to kernel that perform some action on the process behalf.
* The kernel makes a range of services accessible to programs via the system call application programming interface (API). These services include, for example, creating a new process, performing I/O, and creating a pipe for inter process communication.
* A system call changes the processor state from user mode to kernel mode, so that the CPU can access protected kernel memory.

**2-) Select one type of the system calls that you wish to work, and try to explain the main purpose and the functionalities of this system call provides to the operating system, etc.**

We selected the file management type. In file management type, there is a system call that is called mkdir. It is required to give specific path. Then, it automatically creates path with given input.

**3-) As the last step, look over the source code of the system call that you have selected, and try to explain the implementation procedures of this system call as much as possible.**

|  |
| --- |
| #include <string.h> |
|  | #include <limits.h> /\* PATH\_MAX \*/ |
|  | #include <sys/stat.h> /\* mkdir(2) \*/ |
|  | #include <errno.h> |
|  |  |
|  | int mkdir\_p(const char \*path) |
|  | { |
|  | /\* Adapted from http://stackoverflow.com/a/2336245/119527 \*/ |
|  | const size\_t len = strlen(path); |
|  | char \_path[PATH\_MAX]; |
|  | char \*p; |
|  |  |
|  | errno = 0; |
|  |  |
|  | /\* Copy string so its mutable \*/ |
|  | if (len > sizeof(\_path)-1) { |
|  | errno = ENAMETOOLONG; |
|  | return -1; |
|  | } |
|  | strcpy(\_path, path); |
|  |  |
|  | /\* Iterate the string \*/ |
|  | for (p = \_path + 1; \*p; p++) { |
|  | if (\*p == '/') { |
|  | /\* Temporarily truncate \*/ |
|  | \*p = '\0'; |
|  |  |
|  | if (mkdir(\_path, S\_IRWXU) != 0) { |
|  | if (errno != EEXIST) |
|  | return -1; |
|  | } |
|  |  |
|  | \*p = '/'; |
|  | } |
|  | } |
|  |  |
|  | if (mkdir(\_path, S\_IRWXU) != 0) { |
|  | if (errno != EEXIST) |
|  | return -1; |
|  | } |
|  |  |
|  | return 0; |
|  | } |

It is C code. It includes different libraries such as: <limits.h>, <sys/stat.h>, <errno.h> command takes one input first one is char pointe. After taking necessary info such as where to create this directory. It will automatically create it.

**Tugberk GOC 115200084**

**Burak DEMIREL 115200034**