UNIVERSITY OF ENGINEERING AND TECHNOLOGY, TAXILA

FACULTY OF TELECOMMUNICATION AND INFORMATION ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

Operating systems

SYSTEM CALLS II

LAB MANUAL 6

Date:	
Name:	
Reg#:	**Group:**
Marks:	**Signature:**

Operating systems

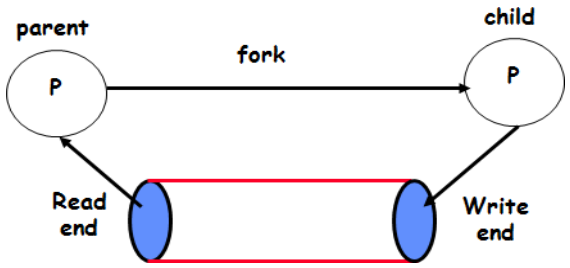
Lab Objective: to understand:

- inter process communication using pipe
- create() time() and ctime()

Inter process communication
IPC facility provides two operations: Send (message) message size fixed or variable, Receive (message). If P and Q wish to communicate, they need to establish a communication link between them and exchange messages via send/receive

Pipe()

Important system calls: **pipe, read, write, close**
pipe: Create a pipe for IPC
read: Read from a pipe
write: Write data to a pipe
close: Close/destroy a pipe



Implementation of pipe()

```
/* Parent creates pipe, forks a child, child writes into
   pipe, and parent reads from pipe */
#include <stdio.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <string.h>
main()
{
        int pipefd[2], pid, n, rc, nr, status;
        char *testString = "Hello, world!\n", buf[1024];
        rc = pipe (pipefd);
        if (rc < 0) {
                perror("pipe");
        }
        pid = fork ();
```

```
        if (pid < 0) {
                perror("fork");
        }
   if (pid == 0) {  /* Child's Code */
       close(pipefd[0]);
       write(pipefd[1], testString, strlen(testString));
       close(pipefd[1]);

   }
    /* Parent's Code */
else
   close(pipefd[1]);
    n = strlen(testString);
    nr = read(pipefd[0], buf, n);
    rc = write(1, buf, nr);
    wait(&status);
    printf("Good work child!\n");
    return(0);
}
```

create()

The prototype for the creat() system call is:

int creat(file_name, mode)

Example

```
#include <stdio.h>
#include <sys/stat.h>         /* defines S_IREAD &amp; S_IWRITE
*/
      int main()
      {
         int fd;
         fd = creat("datafile.dat", S_IREAD | S_IWRITE);
         if (fd == -1)
            printf("Error in opening datafile.dat\n");
         else
            {
            printf("datafile.dat opened for read/write access\n");
            printf("datafile.dat is currently empty\n");
            }
         close(fd);

      }
```

time()

An example is:

/* my_date.c print the current date and time in a format similar to the output of the date command. */

```
#include <stdio.h>
#include <time.h>
void main()
{
time_t rawtime;
time(&rawtime);
printf(" Current time and date: %s",ctime(&rawtime));
}
```

Lab Task: Compute the Fibonacci series using inter process communication through pipe().