



OPERATING SYSTEMS THEORY

ASSIGNMENT #02

GROUP MEMBERS

**USMAN SALMAN – BSEE19045
SAAD RAHIM KHAN – BSEE19031**

TASK:01

Output:

```
(usmansalman@NoOne) - [~/Desktop/assignment02]
$ gcc -o task01 task01.c

(usmansalman@NoOne) - [~/Desktop/assignment02]
$ ./task01
Starting Time(seconds): 1665060626      Starting Time(Micro-seconds): 465089
Ending Time(seconds): 1665060626      Ending Time(Micro-seconds): 465820

Final overhead time(Micro-seconds): 0.073100
```

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <time.h>
#include <sys/time.h>

int main()
{
    struct timeval tv; struct timezone tz;
    gettimeofday(&tv, &tz);
    printf("Starting Time(seconds): %lu\t", tv.tv_sec);
    printf("Starting Time(Micro-seconds): %d\n", tv.tv_usec);
    for(int i=0; i<10000; i++)
    {
        char *temp = (char*) malloc(sizeof(char));
        free(temp);
    }

    struct timeval tv1; struct timezone tz1;
    gettimeofday(&tv1, &tz1);
    //time_t startTime = time(NULL);
    printf("Ending Time(seconds): %lu\t", tv1.tv_sec);
    printf("Ending Time(Micro-seconds): %d\n", tv1.tv_usec);
    int finalTime = tv1.tv_usec - tv.tv_usec;
    printf("\nFinal overhead time(Micro-seconds): %d\n", finalTime);

    return 0;
}
```

TASK: 02

Output:

```
(usmansalman@NoOne)~/Desktop/assignment02]
$ gcc -o task02 task02.c

(usmansalman@NoOne)~/Desktop/assignment02]
$ ./task02

Child Iteration: 0      Child Read: I am Parent      Child Write: I am Child
Parent Iteration: 0     Parent Write: I am Parent    Parent Read: I am Child

Child Iteration: 1      Child Read: I am Parent      Child Write: I am Child
Parent Iteration: 1     Parent Write: I am Parent    Parent Read: I am Child

Child Iteration: 2      Child Read: I am Parent      Child Write: I am Child
Parent Iteration: 2     Parent Write: I am Parent    Parent Read: I am Child

Parent Iteration: 3     Parent Write: I am Parent    Parent Read: I am Child
Child Iteration: 3     Child Read: I am Parent      Child Write: I am Child

Child Iteration: 4      Child Read: I am Parent      Child Write: I am Child
Parent Iteration: 4     Parent Write: I am Parent    Parent Read: I am Child

Child Final overhead time(Micro-seconds): 1503.000000

Parent Final overhead time(Micro-seconds): 1697.000000
```

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <sys/time.h>
int main(void)
{
    int pipefd1[2], pipefd2[2];
    char *Ppin = "I am Parent\n", *Cpin = "I am Child\n";
    int buffer[5] = {}, iterations = 5;
    if(pipe(pipefd1) == -1 || pipe(pipefd2) == -1)
    {
        perror("pipe");
        exit(EXIT_FAILURE);
    }
    struct timeval tv; struct timezone tz;
    gettimeofday(&tv, &tz);
    pid_t pid = fork();
    if(pid == 0)
    {
        for(int i=0; i<iterations; i++)
        {
            printf("\nChild Iteration: %d\t", i);
            read(pipefd2[0], buffer, 11);
            printf("Child Read: %s\t", buffer);
            sleep(1);
            write(pipefd1[1], Cpin, 10);
            printf("\tChild Write: %s\n", Cpin);
            sleep(1);
        }
        struct timeval tv1; struct timezone tz1;
        gettimeofday(&tv1, &tz1);
        float finalTime = tv1.tv_usec - tv.tv_usec;
        printf("\nChild Final overhead time(Micro-seconds): %f\n", finalTime/iterations);
    }
    if(pid > 0)
    {
        for(int i=0; i<iterations; i++)
        {
            printf("Parent Iteration: %d\t", i);
            write(pipefd2[1], Ppin, 11);
            printf("Parent Write: %s\t", Ppin);
            sleep(1);
            read(pipefd1[0], buffer, 10);
            printf("Parent Read: %s\n", buffer);
            sleep(1);
        }
        struct timeval tv1; struct timezone tz1;
        gettimeofday(&tv1, &tz1);
        float finalTime = tv1.tv_usec - tv.tv_usec;
        printf("\nParent Final overhead time(Micro-seconds): %f\n", finalTime/iterations);
    }
    return 0;
}
```

Code Links:

<https://github.com/usman-bsee/Operating-Systems/tree/main/Assignment%2002>

-----*(END)*-----