**Lab 08**

**Lab Assignments:**

1. Complete C programming exercise 2.

**Separating a command line into a sequence of commands**

1. Complete Unix Programming exercise 7.

**Use sigaction**

**Modify the program from Exercise 5 above. Replace the signal function with the sigaction function in the new program.**

Output listing:

root@vishnu-rana:/mnt/c/Users/Vishnu Rana/Desktop/ICT374\_labs\_Rana\_Vishnu/lab08/unix-programming/unix-exercise7# ls

unix-exercise7.c

root@vishnu-rana:/mnt/c/Users/Vishnu Rana/Desktop/ICT374\_labs\_Rana\_Vishnu/lab08/unix-programming/unix-exercise7# gcc unix-exercise7.c

root@vishnu-rana:/mnt/c/Users/Vishnu Rana/Desktop/ICT374\_labs\_Rana\_Vishnu/lab08/unix-programming/unix-exercise7# ./a.out

Sleep call 0

Sleep call 1

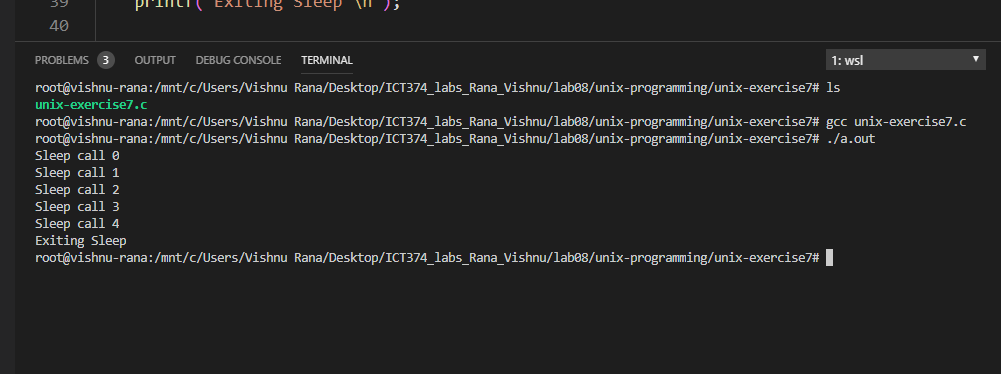
Sleep call 2

Sleep call 3

Sleep call 4

Exiting Sleep

root@vishnu-rana:/mnt/c/Users/Vishnu Rana/Desktop/ICT374\_labs\_Rana\_Vishnu/lab08/unix-programming/unix-exercise7#



1. Complete Unix programming exercise 8.
2. **Catching different signals**

Write a program that will

* 1. create a child process. The child process just sleeps for 10 seconds before it terminates;
  2. set its alarm clock to 20 seconds (in the parent only);
  3. enters an infinite loop. In each loop, it prints one line and then sleep for one second.

However when it (the parent) receives a signal it will respond in the following ways depending on the signal it received:

* 1. SIGCHLD: reports that the child has terminated and prints the child's pid;
  2. SIGALRM: reports that alarm clock has expired.
  3. SIGINT: reports SIGINT is received and prints its numeric value;
  4. any other signal: performs its default action.

Note this program must be written using reliable signal system calls (ie do not use the signal function)

**Output listing:**

root@vishnu-rana:/mnt/c/Users/Vishnu Rana/Desktop/ICT374\_labs\_Rana\_Vishnu/lab08/unix-programming/unix-exercise8# ./a.out sleep 5 7 vfwbr dsfbr

./a.out: invalid time

Process pid=602

Child created

loooping ...

loooping ...

loooping ...

^\loooping ...

loooping ...

^\loooping ...

^\loooping ...

loooping ...

^\loooping ...

^\^\^\loooping ...

^\^\Child expired !loooping ...

^\loooping ...

loooping ...

loooping ...

^CSignal no 2 is caught

loooping ...

^CSignal no 2 is caught

loooping ...

^CSignal no 2 is caught

