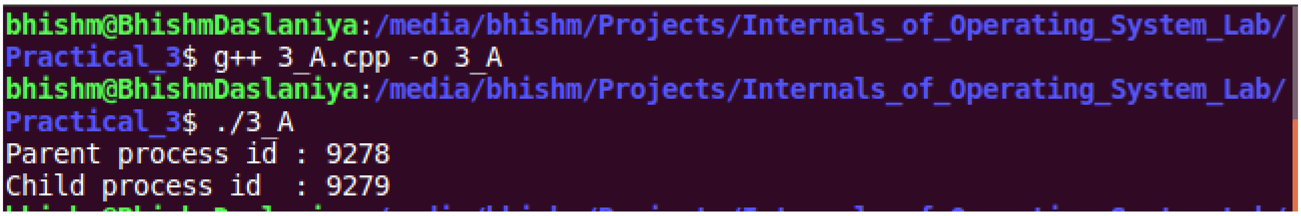
Practical - 3

Aim [A] : Write a C program that will print parent process id and child process id. Mention error checking if child process is not created.

Code :

```
#include<bits/stdc++.h>
#include<stdio.h>
#include<unistd.h>
#include<stdlib.h>
#include<stdarg.h>
#include<string.h>
#include<sys/types.h>
#include <sys/stat.h>
#include<fcntl.h>
#include <dirent.h>

using namespace std;

int main(){
        int pid = fork();
        if(pid < 0){
                perror("Child  process not created!");
                return -1;
        }else if(pid == 0){
                cout<<"Parent process id : "<<getppid()<<endl;
                cout<<"Child process id  : "<<getpid()<<endl;
                return 0;
        }
}
```

Output :



```
bhishm@BhishmDaslaniya:/media/bhishm/Projects/Internals_of_Operating_System_Lab/
Practical_3$ g++ 3_A.cpp -o 3_A
bhishm@BhishmDaslaniya:/media/bhishm/Projects/Internals_of_Operating_System_Lab/
Practical_3$ ./3_A
Parent process id : 9278
Child process id  : 9279
```

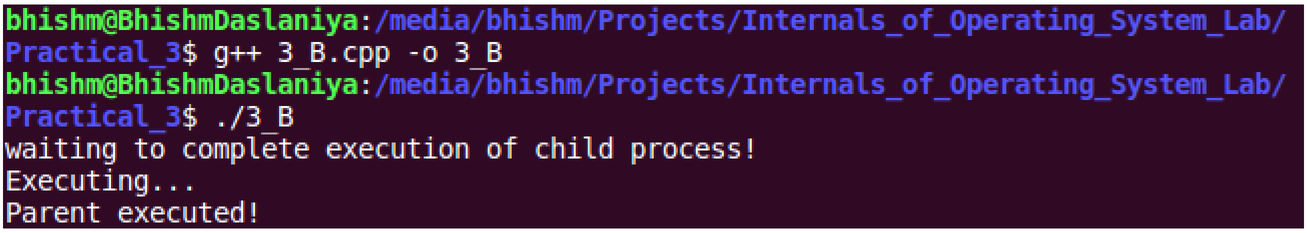
Aim [B] : In continuation of part (a), write a C program where parent process wait for child process to terminate.

Code :

```
#include<bits/stdc++.h>
#include<stdio.h>
#include<unistd.h>
#include<stdlib.h>
#include<stdarg.h>
#include<string.h>
#include<sys/types.h>
#include <sys/stat.h>
#include<fcntl.h>
#include <dirent.h>
```

```
using namespace std;

int main(){
	int pid = fork();
	if(pid < 0){
		perror("Child  process not created!");
		return -1;
	}else if(pid == 0){
		cout<<"Executing..."<<endl;
		return 0;
	}else{
		cout<<"waiting to complete execution of child process!"<<endl;
		sleep(5);
		wait();
		cout<<"Parent executed!"<<endl;
	}
}
```

Output :



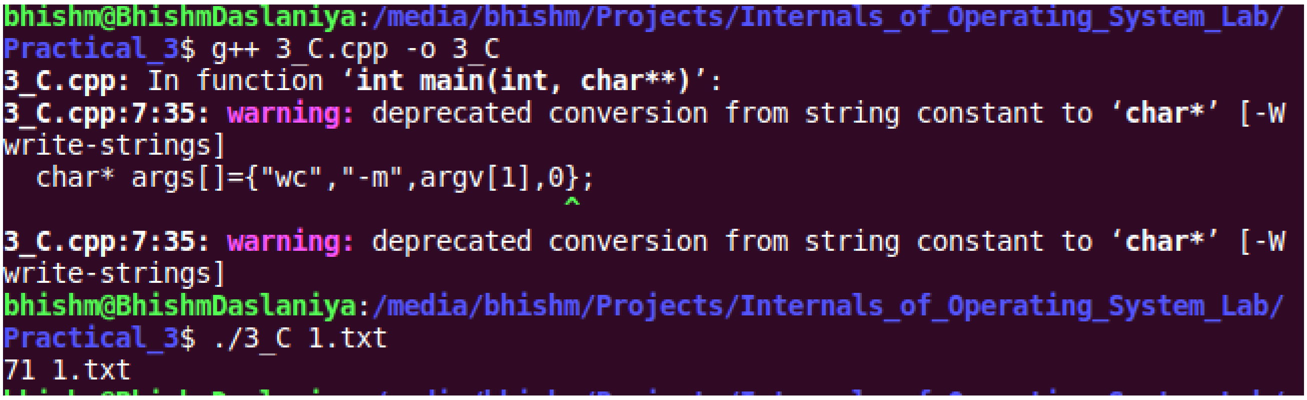
```
bhishm@BhishmDaslaniya:/media/bhishm/Projects/Internals_of_Operating_System_Lab/
Practical_3$ g++ 3_B.cpp -o 3_B
bhishm@BhishmDaslaniya:/media/bhishm/Projects/Internals_of_Operating_System_Lab/
Practical_3$ ./3_B
waiting to complete execution of child process!
Executing...
Parent executed!
```

Aim[C] : Write a C program using execvp() system call which will count the characters from file ‘wc’, using program ‘p.c’.

Code :

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h>
int main(int argc, char*  argv[]){
	char* args[]={"wc","-m",argv[1],0};
	execv("/usr/bin/wc",args);
}
```

Output :



```
bhishm@BhishmDaslaniya:/media/bhishm/Projects/Internals_of_Operating_System_Lab/
Practical_3$ g++ 3_C.cpp -o 3_C
3_C.cpp: In function ‘int main(int, char**)’:
3_C.cpp:7:35: warning: deprecated conversion from string constant to ‘char*’ [-W
write-strings]
  char* args[]={"wc","-m",argv[1],0};
                                   ^

3_C.cpp:7:35: warning: deprecated conversion from string constant to ‘char*’ [-W
write-strings]
bhishm@BhishmDaslaniya:/media/bhishm/Projects/Internals_of_Operating_System_Lab/
Practical_3$ ./3_C 1.txt
71 1.txt
```

Assignment :
Write a C program that will sleep the process for 5000 seconds. Run the same program for three times sequentially. Note down your observations for following questions.

Code:

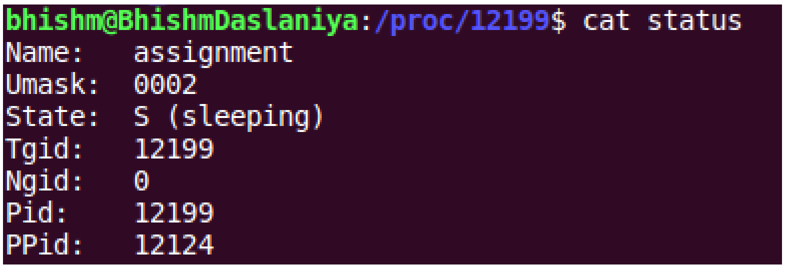
```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h>

int main(int argc, char*  argv[]){
        sleep(5000);
        return 0;
}
```

a. Every time you execute the program, does that process have the same process id?
Ans : No

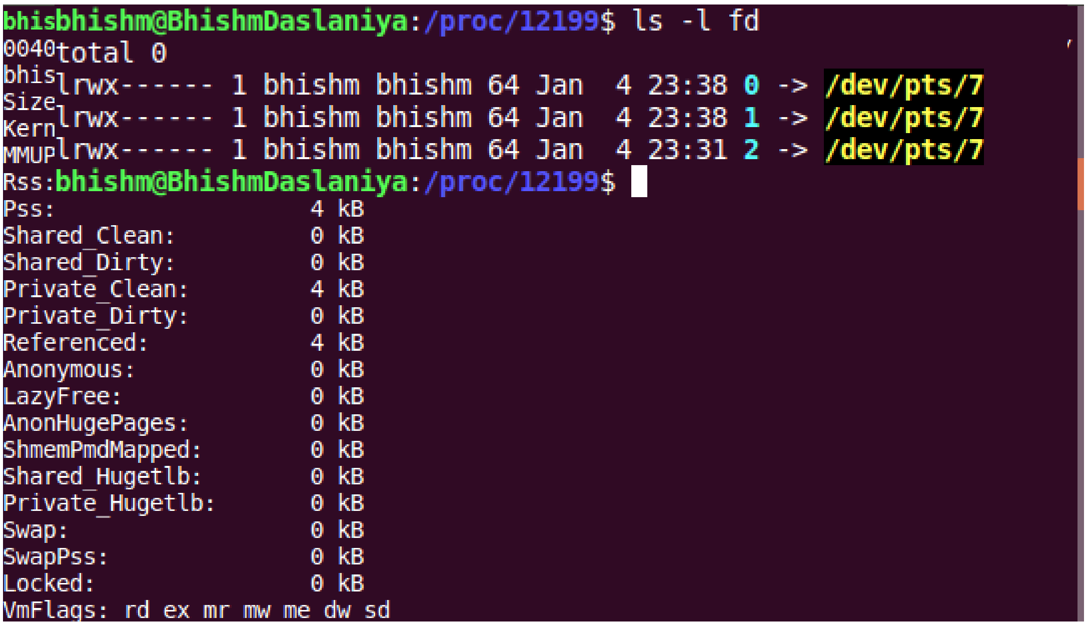
b. Note down following characteristics of process control block using proc and top command.

- Process state
- Process number
- Program counter
- Memory limits
- Open files list
- Number of voluntary context

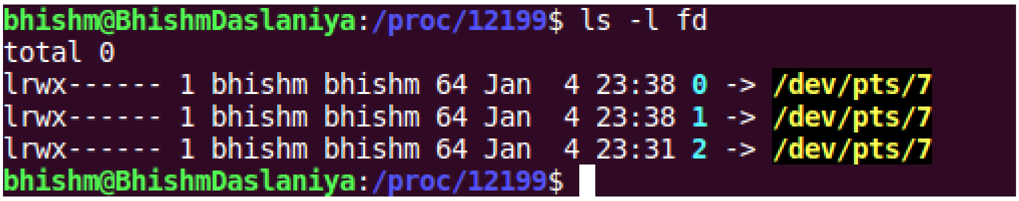
Output :



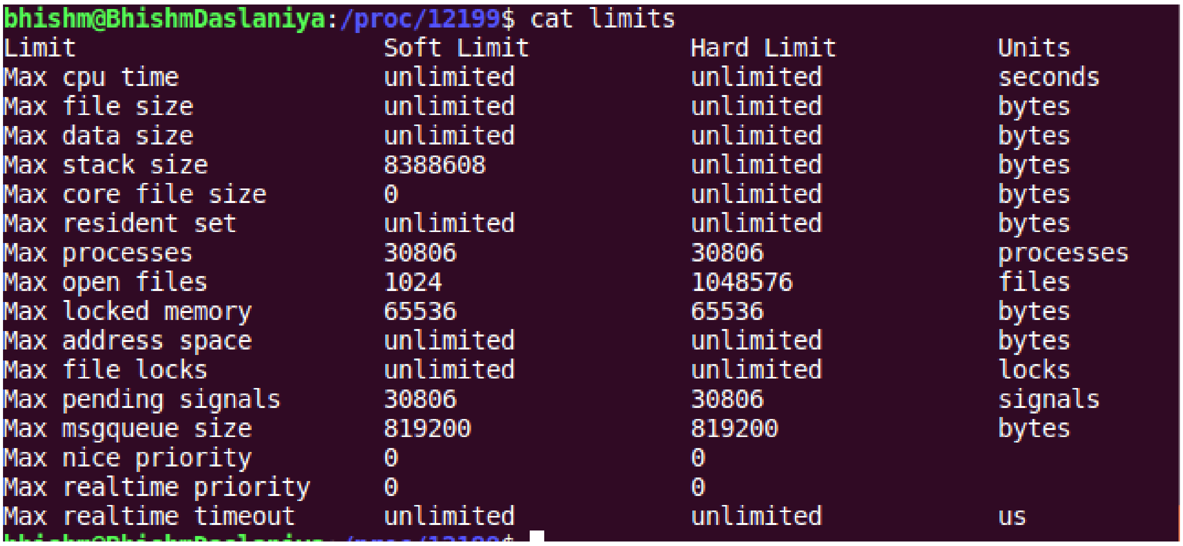
```
bhishm@BhishmDaslaniya:/proc/12199$ cat status
Name:   assignment
Umask:  0002
State:  S (sleeping)
Tgid:   12199
Ngid:   0
Pid:    12199
PPid:   12124
```



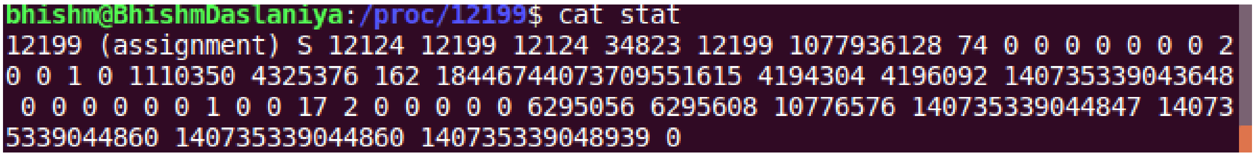
```
bhisbhishm@BhishmDaslaniya:/proc/12199$ ls -l fd
0040total 0
bhislrwx------ 1 bhishm bhishm 64 Jan  4 23:38 0 -> /dev/pts/7
Sizelrwx------ 1 bhishm bhishm 64 Jan  4 23:38 1 -> /dev/pts/7
Kernlrwx------ 1 bhishm bhishm 64 Jan  4 23:38 1 -> /dev/pts/7
MMUPlrwx------ 1 bhishm bhishm 64 Jan  4 23:31 2 -> /dev/pts/7
Rss:bhishm@BhishmDaslaniya:/proc/12199$ 
Pss:                   4 kB
Shared_Clean:          0 kB
Shared_Dirty:          0 kB
Private_Clean:         4 kB
Private_Dirty:         0 kB
Referenced:            4 kB
Anonymous:             0 kB
LazyFree:              0 kB
AnonHugePages:         0 kB
ShmemPmdMapped:        0 kB
Shared_Hugetlb:        0 kB
Private_Hugetlb:       0 kB
Swap:                  0 kB
SwapPss:               0 kB
Locked:                0 kB
VmFlags: rd ex mr mw me dw sd
```



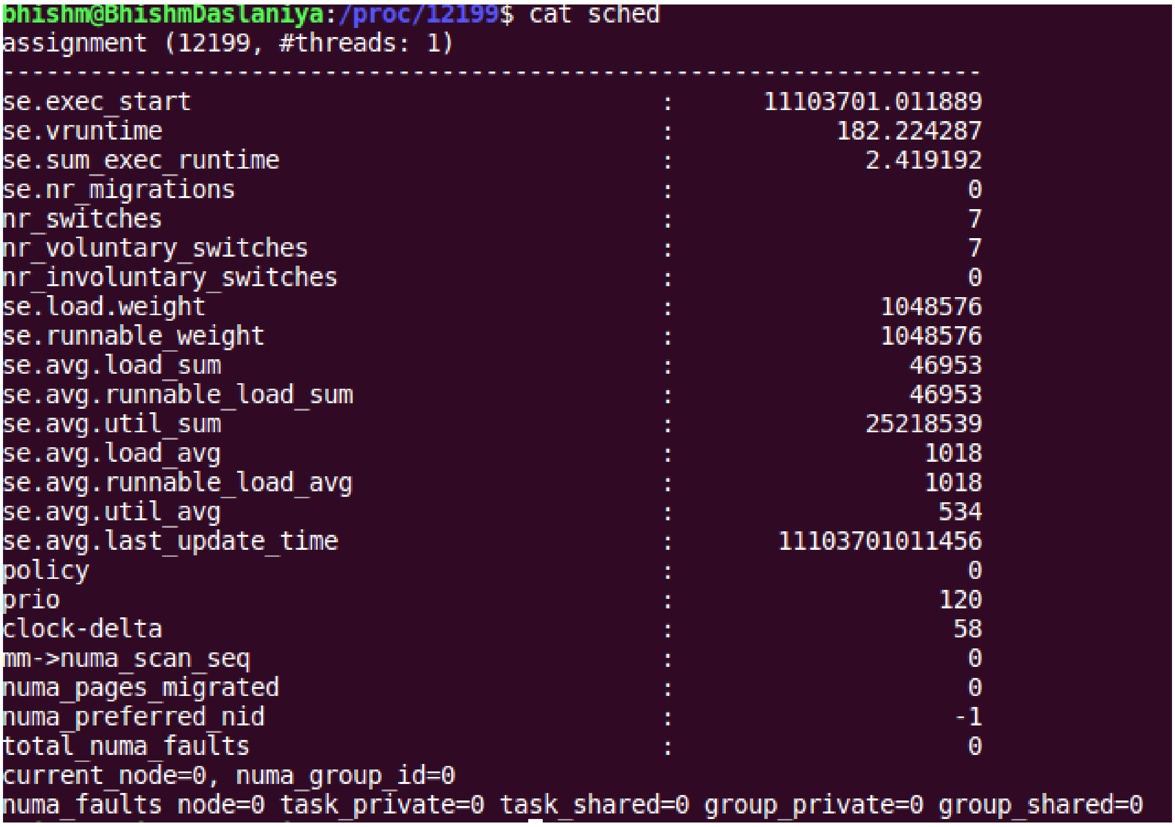
```
bhishm@BhishmDaslaniya:/proc/12199$ ls -l fd
total 0
lrwx------ 1 bhishm bhishm 64 Jan  4 23:38 0 -> /dev/pts/7
lrwx------ 1 bhishm bhishm 64 Jan  4 23:38 1 -> /dev/pts/7
lrwx------ 1 bhishm bhishm 64 Jan  4 23:31 2 -> /dev/pts/7
bhishm@BhishmDaslaniya:/proc/12199$
```



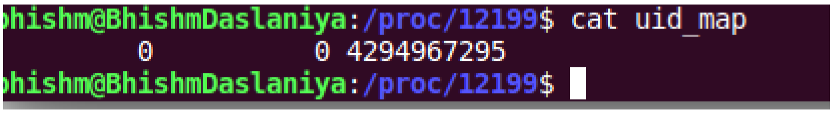
```
bhishm@BhishmDaslaniya:/proc/12199$ cat limits
Limit                     Soft Limit           Hard Limit           Units
Max cpu time              unlimited            unlimited            seconds
Max file size             unlimited            unlimited            bytes
Max data size             unlimited            unlimited            bytes
Max stack size            8388608              unlimited            bytes
Max core file size        0                    unlimited            bytes
Max resident set          unlimited            unlimited            bytes
Max processes             30806                30806                processes
Max open files            1024                 1048576              files
Max locked memory         65536                65536                bytes
Max address space         unlimited            unlimited            bytes
Max file locks            unlimited            unlimited            locks
Max pending signals       30806                30806                signals
Max msgqueue size         819200               819200               bytes
Max nice priority         0                    0
Max realtime priority     0                    0
Max realtime timeout      unlimited            unlimited            us
bhishm@BhishmDaslaniya:/proc/12199$
```



```
bhishm@BhishmDaslaniya:/proc/12199$ cat stat
12199 (assignment) S 12124 12199 12124 34823 12199 1077936128 74 0 0 0 0 0 0 0 2
0 0 1 0 1110350 4325376 162 18446744073709551615 4194304 4196092 140735339043648
 0 0 0 0 0 0 1 0 0 17 2 0 0 0 0 0 6295056 6295608 10776576 140735339044847 14073
5339044860 140735339044860 140735339048939 0
```



```
bhishm@BhishmDaslaniya:/proc/12199$ cat sched
assignment (12199, #threads: 1)
-------------------------------------------------------------------
se.exec_start                                :      11103701.011889
se.vruntime                                  :           182.224287
se.sum_exec_runtime                          :             2.419192
se.nr_migrations                             :                    0
nr_switches                                  :                    7
nr_voluntary_switches                        :                    7
nr_involuntary_switches                      :                    0
se.load.weight                               :              1048576
se.runnable_weight                           :              1048576
se.avg.load_sum                              :                46953
se.avg.runnable_load_sum                     :                46953
se.avg.util_sum                              :             25218539
se.avg.load_avg                              :                 1018
se.avg.runnable_load_avg                     :                 1018
se.avg.util_avg                              :                  534
se.avg.last_update_time                      :       11103701011456
policy                                       :                    0
prio                                         :                  120
clock-delta                                  :                   58
mm->numa_scan_seq                            :                    0
numa_pages_migrated                          :                    0
numa_preferred_nid                           :                   -1
total_numa_faults                            :                    0
current_node=0, numa_group_id=0
numa_faults node=0 task_private=0 task_shared=0 group_private=0 group_shared=0
```



```
bhishm@BhishmDaslaniya:/proc/12199$ cat uid_map
         0          0 4294967295
bhishm@BhishmDaslaniya:/proc/12199$
```