Naron Chatjitakornkul   6530113921
Pongpak Manoret   6532126421
Pupipat Singkhorn   6532142421
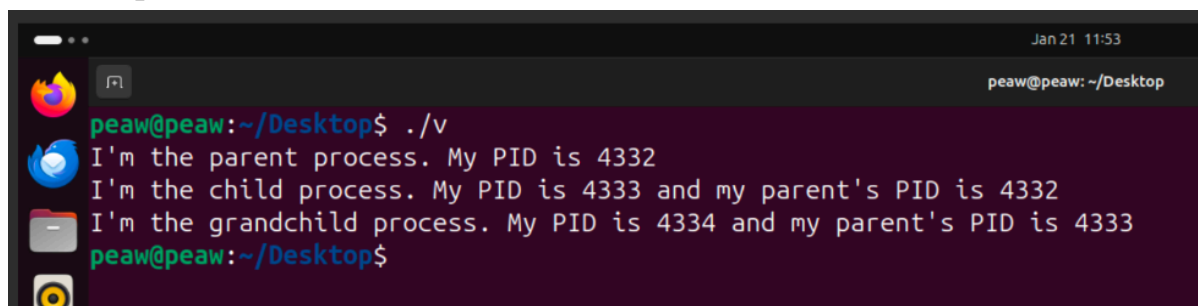
1. Source code:

```c
peaw@peaw:~/Desktop$ cat sol1.c
#include <stdio.h>
#include <unistd.h>

int main() {
    pid_t pid = fork();

    if (pid == -1) {
        perror("fork failed");
        return 1;
    } else if (pid == 0) {
        // Child process
        pid_t pid1 = fork();
        if(pid1 == 0){
            printf("I'm the grandchild process. My PID is %d and my parent's PID is %d\n", getpid(),getppid());
            wait(NULL);
        }
        else{
            printf("I'm the child process. My PID is %d and my parent's PID is %d\n", getpid(),getppid());
            wait(NULL);
        }
    } else {
        // Parent process
        printf("I'm the parent process. My PID is %d\n", getpid());
        wait(NULL);
    }

    return 0;
}
peaw@peaw:~/Desktop$ ###
```

Output

```
peaw@peaw:~/Desktop$ ./v
I'm the parent process. My PID is 4332
I'm the child process. My PID is 4333 and my parent's PID is 4332
I'm the grandchild process. My PID is 4334 and my parent's PID is 4333
peaw@peaw:~/Desktop$
```
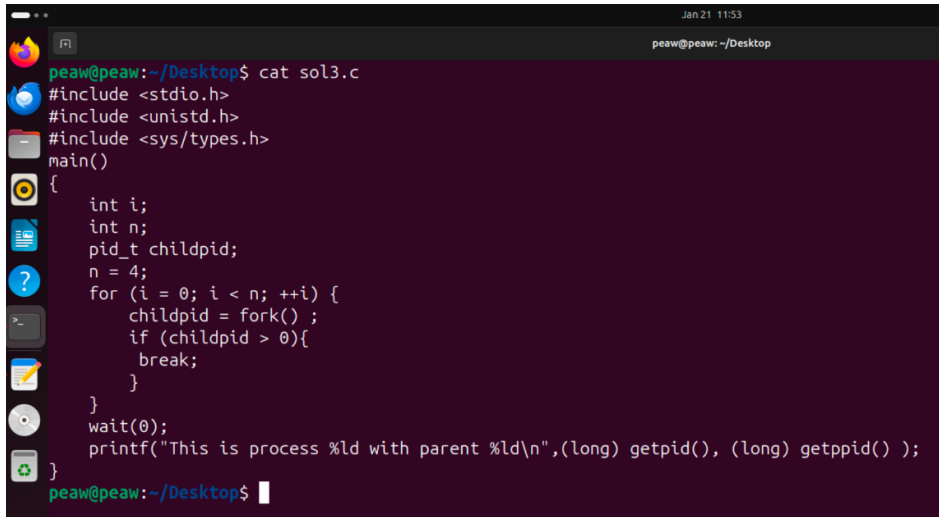
2. Source code:

```
naron@DESKTOP-00OOCNB: ~
  GNU nano 6.2
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
int compute_period;
int sleep_period;
int i;
/* what to do when alarm is on */
void on_alarm (int signal) {
    printf("Sleep\n");
    sleep (sleep_period);
    printf ("Wake up\n");
    /* activate alarm again */
    alarm (compute_period);
}
main ()
{
    printf("Enter compute period :\n");
    scanf("%d", &compute_period);
    printf("Enter sleep period :\n");
    scanf("%d", &sleep_period);
    int i;
    /* on_alarm() is signal handler for SIGALARM */
    signal(SIGALRM, on_alarm);
    /* activate alarm */
    alarm (compute_period);
    /* compute infinitely but can be interrupted by alarm */
    for (i=0; ;i++) {
        if (i==0) printf("computing\n");
    }
}
```
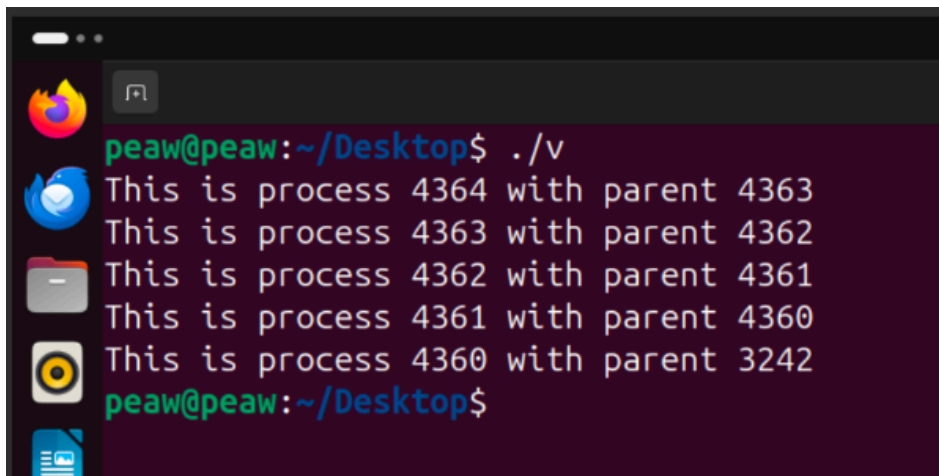
Output:

```
naron@DESKTOP-00OOCNB:~$ ./ac32
Enter compute period :
1
Enter sleep period :
1
computing
Sleep
Wake up
Sleep
Wake up
Sleep
Wake up
```

3. Source code

```
peaw@peaw:~/Desktop$ cat sol3.c
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
main()
{
    int i;
    int n;
    pid_t childpid;
    n = 4;
    for (i = 0; i < n; ++i) {
        childpid = fork() ;
        if (childpid > 0){
         break;
        }
    }
    wait(0);
    printf("This is process %ld with parent %ld\n",(long) getpid(), (long) getppid() );
}
peaw@peaw:~/Desktop$
```

Output:

```
peaw@peaw:~/Desktop$ ./v
This is process 4364 with parent 4363
This is process 4363 with parent 4362
This is process 4362 with parent 4361
This is process 4361 with parent 4360
This is process 4360 with parent 3242
peaw@peaw:~/Desktop$
```