***OS Lab Week 9***

RA1911026010037

Kinjalk Parth

Exp 9:

Example.c:

example.c

#include<stdio.h>

#include<unistd.h>

int main()

{

int p\_id;

p\_id=getpid();

printf("Process ID of example.c: %d\n",p\_id);

printf("\n");

char \*args[]={"./hello", NULL};

execv(args[0],args);

}

Hello.c:

#include<stdio.h>

#include<unistd.h>

int main()

{

int p\_id;

p\_id=getpid();

printf("We are in hello.c\n");

printf("Process ID of hello.c: %d\n",p\_id);

printf("\n");

}

Output:

Exp10:

Example2

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

#include<sys/wait.h>

int main(int argc,char \*argv[])

{

printf("Process id of example2.c = %d\n",getpid());

pid\_t pid;

pid=fork();

if(pid>0)

{

printf("\nThe control is in parent process\n");

printf("Process id of parent = %d\n",getpid());

}

else

{

printf("\nThe control is in child process\n");

printf("Process id of child = %d\n",getpid());

printf("\nCalling hello2.c from child\n");

char \*arg[]={NULL};

execv("./hello2",arg);

}

wait(NULL);

return 0;

}

child.c

#include<stdio.h>

#include<unistd.h>

#include<stdlib.h>

int main(int argc,char \*argv[])

{

printf("\nWe are in hello2.c");

printf("\nProcess id of hello2.c = %d\n",getpid());

return 0;

}

Output:



