Day:20/05/2025

Code:

```
vim parentpp.c
#include<stdio.h>
#include<unistd.h>

int main(){
  int f=fork();
  if(f==0){
    printf("I am a child My parent id is %d\n",getppid());
    printf("My id is %d\n",getpid());
  }

  else{
    int f1=fork();
     if(f1==0){
       printf("I am sibling. My parent id is %d\n",getppid());
       printf("My id is %d\n",getpid());
     }
    }


  return 0;
}
```

Output:

```
{ ~ }   » vi parentpp.c
{ ~ }   » gcc parentpp.c -o parentpp
{ ~ }   » ./parentpp
I am a child My parent id is 12044
My id is 2100
I am sibling. My parent id is 12044
My id is 13848
{ ~ }   » |
```

Conclusions: #include<unistd.h>-header files containing system calls

fork() -creating a new child process

This fork() returns 0 to the child

getppid()-process id of its parent

getpid()-process id of the child it self