```
#include<stdio.h>
void main()
{
int pid;
pid = fork();
if(pid > 0)
{
printf (" Iam parent\n");
}
else
{
printf("Iam child\n");
}
}
```

The parent process prints the first statement and the child prints the next statement.

3. exec

New programs can be run using exec system calls. When a process calls exec, the process is completely replaced by the new program. The new program starts executing from its main function.

A new process is not created, process id remains the same, and the current process's text, data, heap, and stack segments are replaced by the new program. exec has many flavors one of which is execv.

execv takes two parameters. The first is the pathname of the program that is going to be executed. The second is a pointer to an array of pointers that hold the addresses of arguments. These arguments are the command line arguments for the new program.

4. wait

When a process terminates, its parent should receive some information regarding the process like the process id, the termination status, amount of CPU time taken etc. This is possible only if the parent process waits for the termination of the child process. This waiting is done by calling the wait system call. When the child process is running, the parent blocks when wait is called. If the child terminates normally or abnormally, wait immediately returns with the termination status of