

Ejemplos procesos



Julia Clemente Párraga

/gso>

Ejemplo 1

```
int main()
{
    int ident, a = 1;

    if ((ident = fork()) == -1) {
        perror("Error en fork");
        exit(1);
    }
    else if (ident != 0) {
        a = a + 3;
        printf("%d\n", a);
    }
    else {
        a = a * 3;
        printf("%d\n", a);
    }
    return 0;
}
```

Ejemplo 2

```
int main()
{
    int n;

    if ((n = fork()) == -1)
    {
        printf("Error en creación de proceso\n");
        exit(1);
    }
    else if ( n  == 0 )
        printf("Identificador de proceso: %d\n", getpid());
    else
    {
        printf("Valor de fork: %d\n", n);
        printf("Identificador del proceso: %d\n", getpid());
    }
    return 0;
}
```

Ejemplo 3

```
int main()
{
    pid_t id;
    int i;
    int nproc = 5;

    for ( i = 0; i < nproc; i++)
    {
        id = fork();
        if (id != 0)
            break;
    }
    printf("El padre del proceso %d es %d\n", getpid(), getppid());
    getchar();
    return 0;
}
```

Ejemplo 4

```
int main()
{
    pid_t id;
    int i;
    int nproc = 5;

    for ( i = 0; i < nproc; i++)
    {
        id = fork();
        if (id == 0)
            break;
    }

    printf("El padre del proceso %d es %d\n", getpid(), getppid());
    return 0;
}
```

Ejemplo 5

```
int main(int argc, char **argv)
{
    pid_t id;

    id = fork();
    switch(id) {
        case -1:
            exit(-1);
        case 0:
            if (execvp(argv[1], &argv[1]) < 0)
                perror("Error en exec");
            default:
                printf("Fin de proceso padre o hijo?\n");
    }
    return 0;
}
```

Ejemplo 6

```
int main(int argc, char **argv)
{
    int n;

    if ((n = fork()) == -1)
    {
        printf("Error en fork()\n");
        exit(1);
    }
    else if (n == 0) {
        printf("Hijo\n");
    }
    else {
        printf("Padre\n");
        wait(NULL);
        printf("Ya terminó el hijo\n");
    }
    return 0;
}
```

Ejemplo 7

```
int main(int argc, char **argv)
{
    pid_t id;

    id = fork();
    if (id == -1)
        exit(-1);
    else if (id == 0) {
        if (execvp(argv[1], &argv[1]) < 0)
            perror("Error en exec");
    }
    else
        exit(0);
}
```


Ejemplo 8

```
int main(int argc, char **argv)
{
    pid_t id, antppid, postppid;
    pid_t pidpadre;

    pidpadre = getpid();
    id = fork();
    if (id == -1) {
        perror("Error en creación de proceso");
        exit(1);
    }
}
```

Ejemplo 8 (cont.)

```
else if (id == 0) {
    antppid = getppid();
    sleep(3);
    postppid = getppid();
    printf("El padre inicial del hijo es %d\n", pidpadre);
    printf("El hijo creado es %d\n", getpid());
    printf("El anterior ppid del hijo es %d\n", antppid);
    printf("El ppid actual del hijo es %d\n", postppid);
    exit(0);
}
else {
    sleep(1);
    exit(0);
}
}
```