



# **FORK – EXEMPEL**

```
static int g = 10;
int main(void) {
    int v = 10;
    pid_t p;
    while (++v > 8)
        if ((p = fork()) < 0) {
            perror("fork error");
            return 1;
        } else if (p == 0) {
            v-=2;
            g++;
        } else {
            g--;
            v-=3;
            if (waitpid(p, NULL, 0) != p) {
                perror("waitpid error");
                return 1;
            }
        }
    printf("mypid = %d parentpid = %d p = %d v = %d g = %d\n",
           getpid(), getppid(), p, v, g);
    return 0;
}
```



PID: 100

g: 10

v: 10

p: ?



v++

PID: 100

g: 10


v: ~~10~~ 11

p: ?

fork

PID: 100  
g: 10  
v: ~~10~~ 11  
p: 101

PID: 101  
g: 10  
v: 11  
p: 0



The diagram illustrates a process state transition. It consists of two rectangular boxes. The top box has a green border and contains the text: PID: 100, g: 10, v: ~~10~~ 11, and p: 101. The bottom box has a yellow border and contains the text: PID: 101, g: 10, v: 11, and p: 0. A black arrow points from the bottom of the green box to the top of the yellow box, indicating a transition from the state in the green box to the state in the yellow box.

PID: 100

g: 10

v: ~~10~~ 11

p: 101

PID: 101

g: 10

v: 11

p: 0

g--

PID: 100

g: ~~10~~ 9

v: ~~10~~ 11

p: 101

PID: 101

g: 10

v: 11

p: 0

v-=3

PID: 100  
g: ~~10~~ 9  
v: ~~10~~ ~~11~~ 8  
p: 101

PID: 101  
g: 10  
v: 11  
p: 0



wait

PID: 100  
g: ~~10~~ 9  
v: ~~10~~ ~~11~~ 8  
p: 101

PID: 101  
g: 10  
v: 11  
p: 0

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

v-=2

```
PID: 101  
g: 10  
v: 11 9  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

g++

```
PID: 101  
g: 10 11  
v: 11 9  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

++v

```
PID: 101  
g: 10 11  
v: 11 9 10  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

fork

```
PID: 101  
g: 10 11  
v: 11 9 10  
p: 0 102
```

```
PID: 102  
g: 11  
v: 10  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

```
PID: 101  
g: 10 11  
v: 11 9 10  
p: 0 102
```

```
PID: 102  
g: 11  
v: 10  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

g-  
v-=3

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11  
v: 10  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11  
v: 10  
p: 0
```



wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11 12  
v: 10 8  
p: 0
```

v-=2  
g++

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11 12  
v: 10 8 9  
p: 0
```

++v

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

fork

```
PID: 102  
g: 11 12  
v: 10 8 9  
p: 0 103
```

```
PID: 103  
g: 12  
v: 9  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11 12  
v: 10 8 9  
p: 0 103
```

```
PID: 103  
g: 12  
v: 9  
p: 0
```



wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6  
p: 0 103
```

g-  
v-=3

```
PID: 103  
g: 12  
v: 9  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

wait

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6  
p: 0 103
```

```
PID: 103  
g: 12  
v: 9  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

wait

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6  
p: 0 103
```

v-=2  
g++

```
PID: 103  
g: 12 13  
v: 9 7  
p: 0
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

wait

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6  
p: 0 103
```

++v

```
PID: 103  
g: 12 13  
v: 9 7 8  
p: 0
```



wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

wait

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6  
p: 0 103
```

printf

```
PID: 103  
g: 12 13  
v: 9 7 8  
p: 0
```

mypid = 103    parentpid = 102    p =    0    v = 8    g = 13

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6  
p: 0 103
```

exit

```
mypid = 103   parentpid = 102   p =    0   v = 8   g = 13
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6 7  
p: 0 103
```

++v

mypid = 103    parentpid = 102    p =    0    v = 8    g = 13

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

wait

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7  
p: 0 102
```

printf

```
PID: 102  
g: 11 12 11  
v: 10 8 9 6 7  
p: 0 103
```

```
mypid = 103   parentpid = 102   p =    0   v =  8   g = 13  
mypid = 102   parentpid = 101   p = 103   v =  7   g = 11
```

wait

PID: 100  
g: ~~10~~ 9  
v: ~~10~~ ~~11~~ 8  
p: 101

PID: 101  
g: ~~10~~ ~~11~~ 10  
v: ~~11~~ 9 ~~10~~ 7  
p: ~~0~~ 102

exit

mypid = 103    parentpid = 102    p =    0    v = 8    g = 13  
mypid = 102    parentpid = 101    p = 103    v = 7    g = 11

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

++v

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7 8  
p: 0 102
```

```
mypid = 103   parentpid = 102   p =    0   v =  8   g = 13  
mypid = 102   parentpid = 101   p = 103   v =  7   g = 11
```

wait

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

printf

```
PID: 101  
g: 10 11 10  
v: 11 9 10 7 8  
p: 0 102
```

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10

```
PID: 100  
g: 10 9  
v: 10 11 8  
p: 101
```

exit

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10

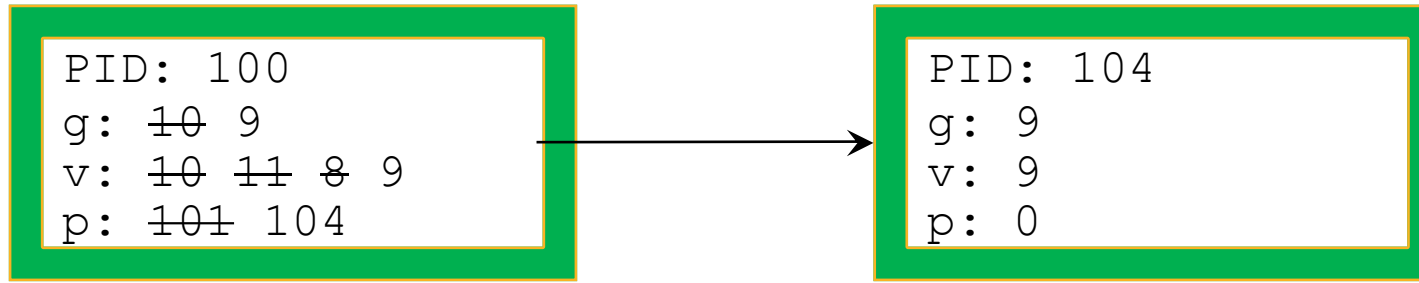


++v

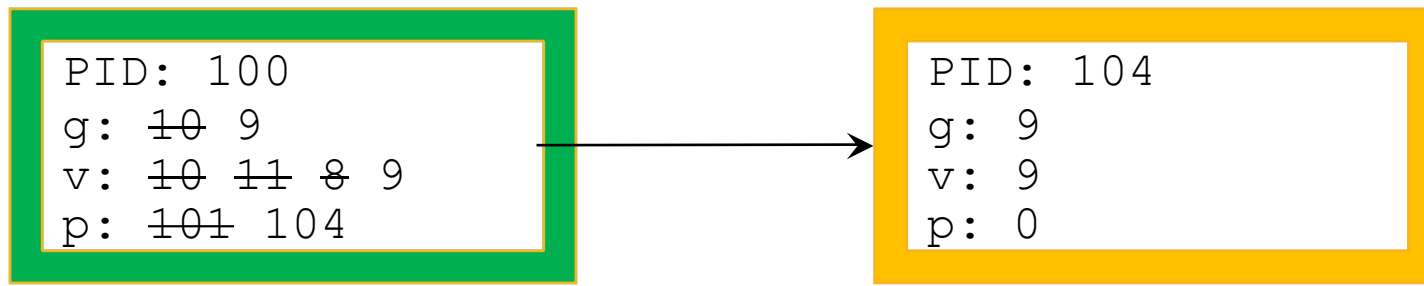
```
PID: 100  
g: 10 9  
v: 10 11 8 9  
p: 101
```

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10

fork



mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10



mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10

g-  
v-=3

PID: 100  
g: ~~10~~ ~~9~~ 8  
v: ~~10~~ ~~11~~ ~~8~~ ~~9~~ 6  
p: ~~101~~ 104

PID: 104  
g: 9  
v: 9  
p: 0

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10

wait

PID: 100  
g: ~~10~~ ~~9~~ 8  
v: ~~10~~ ~~11~~ ~~8~~ ~~9~~ 6  
p: ~~101~~ 104

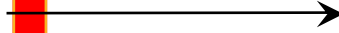
PID: 104  
g: ~~9~~ 10  
v: ~~9~~ 7  
p: 0

v-=2  
g++

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10

wait

PID: 100  
g: ~~10~~ ~~9~~ 8  
v: ~~10~~ ~~11~~ ~~8~~ ~~9~~ 6  
p: ~~101~~ 104



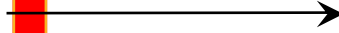
PID: 104  
g: ~~9~~ 10  
v: ~~9~~ 7 8  
p: 0

++v

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10

wait

```
PID: 100  
g: 10 9 8  
v: 10 11 8 9 6  
p: 101 104
```



```
PID: 104  
g: 9 10  
v: 9 7 8  
p: 0
```

printf

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10
mypid = 104	parentpid = 100	p = 0	v = 8	g = 10

PID: 100

g: ~~10~~ ~~9~~ 8

v: ~~10~~ ~~11~~ ~~8~~ ~~9~~ 6

p: ~~101~~ 104

exit

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10
mypid = 104	parentpid = 100	p = 0	v = 8	g = 10



++v

```
PID: 100  
g: 10 9 8  
v: 10 11 8 9 6 7  
p: 101 104
```

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10
mypid = 104	parentpid = 100	p = 0	v = 8	g = 10

printf

```
PID: 100  
g: 10 9 8  
v: 10 11 8 9 6 7  
p: 101 104
```

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10
mypid = 104	parentpid = 100	p = 0	v = 8	g = 10
mypid = 100	parentpid = 59	p = 104	v = 7	g = 8

exit

mypid = 103	parentpid = 102	p = 0	v = 8	g = 13
mypid = 102	parentpid = 101	p = 103	v = 7	g = 11
mypid = 101	parentpid = 100	p = 102	v = 8	g = 10
mypid = 104	parentpid = 100	p = 0	v = 8	g = 10
mypid = 100	parentpid = 59	p = 104	v = 7	g = 8