

fork() step by step...

Process 1000

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {        // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                 // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...
global_var = 3
...

Process's variables

fork() step by step...

Process 1000

```
int global_var = 3;
int main(void)
{
  int local_var = 3;
  pid_t proc_id = fork();
  if(proc_id == -1) exit(1); // Error...
  if(proc_id == 0) {      // Child process
    global_var = 2;
    local_var = 2;
    printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
  } else {                // Parent process
    global_var = 4;
    local_var = 4;
    printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
  }
}
```

...

global_var = 3
local_var = 3

...

Standard output (stdout):

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 3
local_var = 3
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 3
local_var = 3
proc_id = 0
...
```

Standard output (stdout):

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;

    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) { // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else { // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...
global_var = 3
local_var = 3
proc_id = 1001
...

Wrong...

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) { // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else { // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...
global_var = 3
local_var = 3
proc_id = 0
...

Standard output (stdout):

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) { // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else { // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

Wrong...

```
...
global_var = 3
local_var = 3
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) { // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else { // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 3
local_var = 3
proc_id = 0
...
```

Standard output (stdout):

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 3
local_var = 3
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 3
local_var = 3
proc_id = 0
...
```

Standard output (stdout):

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 4
local_var = 3
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 3
local_var = 3
proc_id = 0
...
```

Standard output (stdout):

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

→

```
...
global_var = 4
local_var = 4
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

→

```
...
global_var = 3
local_var = 3
proc_id = 0
...
```

Standard output (stdout):

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 4
local_var = 4
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

```
...
global_var = 3
local_var = 3
proc_id = 0
...
```

Standard output (stdout):

```
PARENT: global_var=4 local_var=4
```

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



...

global_var = 4
local_var = 4
proc_id = 1001

...

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



...

global_var = 3
local_var = 3
proc_id = 0

...

Standard output (stdout):

PARENT: global_var=4 local_var=4

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 4
local_var = 4
proc_id = 1001

...

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 3
local_var = 3
proc_id = 0

...

Wrong...

Standard output (stdout):

PARENT: global_var=4 local_var=4

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 4
local_var = 4
proc_id = 1001

...

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 3
local_var = 3
proc_id = 0

...

Standard output (stdout):

PARENT: global_var=4 local_var=4

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 4
local_var = 4
proc_id = 1001

...

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 2
local_var = 3
proc_id = 0

...

Standard output (stdout):

PARENT: global_var=4 local_var=4

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 4
local_var = 4
proc_id = 1001

...

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```

...

global_var = 2
local_var = 2
proc_id = 0

...


Standard output (stdout):

PARENT: global_var=4 local_var=4

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



```
...
global_var = 4
local_var = 4
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



```
...
global_var = 2
local_var = 2
proc_id = 0
...
```

Standard output (stdout):

```
PARENT: global_var=4 local_var=4
CHILD : global_var=2 local_var=2
```

fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



```
...
global_var = 4
local_var = 4
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



```
...
global_var = 2
local_var = 2
proc_id = 0
...
```

Standard output (stdout):

```
PARENT: global_var=4 local_var=4
CHILD : global_var=2 local_var=2
```


fork() step by step...

Process 1000 (PARENT)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



```
...
global_var = 4
local_var = 4
proc_id = 1001
...
```

Process 1001 (CHILD)

```
int global_var = 3;
int main(void)
{
    int local_var = 3;
    pid_t proc_id = fork();
    if(proc_id == -1) exit(1); // Error...
    if(proc_id == 0) {           // Child process
        global_var = 2;
        local_var = 2;
        printf("CHILD: global_var=%d local_var=%d\n", global_var, local_var);
    } else {                     // Parent process
        global_var = 4;
        local_var = 4;
        printf("PARENT: global_var=%d local_var=%d\n", global_var, local_var);
    }
}
```



```
...
global_var = 2
local_var = 2
proc_id = 0
...
```

Standard output (stdout):

```
PARENT: global_var=4 local_var=4
CHILD : global_var=2 local_var=2
```