

**Princeton University**  
**COS 217: Introduction to Programming Systems**  
**Trace of testforkret**

```
% gcc217 testforkret.c -o testforkret
```

**Princeton University**  
**COS 217: Introduction to Programming Systems**  
**Trace of testforkret**

```
% ./testforkret
```

```
29409
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
            (int)getPid());
    else
        printf("%d parent\n",
            (int)getPid());
    printf("%d parent and child\n",
        (int)getPid());
    return 0;
}
```

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
            (int)getPid());
    else
        printf("%d parent\n",
            (int)getPid());
    printf("%d parent and child\n",
        (int)getPid());
    return 0;
}
```

**Princeton University**  
**COS 217: Introduction to Programming Systems**  
**Trace of testforkret**

```
% ./testforkret
```

```
29409
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
            (int)getPid());
    else
        printf("%d parent\n",
            (int)getPid());
    printf("%d parent and child\n",
        (int)getPid());
    return 0;
}
```

**Writes:**

```
29409 parent
```

**Princeton University**  
**COS 217: Introduction to Programming Systems**  
**Trace of testforkret**

```
% ./testforkret
```

```
29409
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
            (int)getPid());
    else
        printf("%d parent\n",
            (int)getPid());
    printf("%d parent and child\n",
        (int)getPid());
    return 0;
}
```

**Writes:**

```
29409 parent
```

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
            (int)getPid());
    else
        printf("%d parent\n",
            (int)getPid());
    printf("%d parent and child\n",
        (int)getPid());
    return 0;
}
```

Writes:

```
29409 parent
```

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

concurrent

```
29410
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

Assume OS gives CPU to parent

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

29410

```
29410
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

concurrent

Assume OS gives CPU to parent



Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

29410

```
29410
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

concurrent

Assume OS gives CPU to parent

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

29410

```
29410
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

concurrent

Writes:

29409 parent

Assume OS gives CPU to parent

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

29410

```
29410
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int)getPid());
    else
        printf("%d parent\n",
               (int)getPid());
    printf("%d parent and child\n",
           (int)getPid());
    return 0;
}
```

concurrent

Writes:

29409 parent and child

Assume OS gives CPU to parent

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29409
int main(void)
{
  pid_t iPid;
  printf("%d parent\n",
        (int) getpid());
  fflush(stdin);
  fflush(stdout);
  iPid = fork();
  if (iPid == 0)
    printf("%d child\n",
          (int) getpid());
  else
    printf("%d parent\n",
          (int) getpid());
  printf("%d parent and child\n",
        (int) getpid());
  return 0;
}
```

29410

```
29410
int main(void)
{
  pid_t iPid;
  printf("%d parent\n",
        (int) getpid());
  fflush(stdin);
  fflush(stdout);
  iPid = fork();
  if (iPid == 0)
    printf("%d child\n",
          (int) getpid());
  else
    printf("%d parent\n",
          (int) getpid());
  printf("%d parent and child\n",
        (int) getpid());
  return 0;
}
```

concurrent

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29410
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int) getpid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int) getpid());
    else
        printf("%d parent\n",
               (int) getpid());
    printf("%d parent and child\n",
           (int) getpid());
    return 0;
}
```

```
0
```

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29410
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int) getpid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int) getpid());
    else
        printf("%d parent\n",
               (int) getpid());
    printf("%d parent and child\n",
           (int) getpid());
    return 0;
}
```

0

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

29410

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int) getpid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int) getpid());
    else
        printf("%d parent\n",
               (int) getpid());
    printf("%d parent and child\n",
           (int) getpid());
    return 0;
}
```

0

Writes:  
29410 child

Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
29410
```

```
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int) getpid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int) getpid());
    else
        printf("%d parent\n",
               (int) getpid());
    printf("%d parent and child\n",
           (int) getpid());
    return 0;
}
```

0

Writes:

29410 parent and child



Princeton University  
COS 217: Introduction to Programming Systems  
Trace of testforkret

```
% ./testforkret
```

```
19410
int main(void)
{
    pid_t iPid;
    printf("%d parent\n",
           (int) getpid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
        printf("%d child\n",
               (int) getpid());
    else
        printf("%d parent\n",
               (int) getpid());
    printf("%d parent and child\n",
           (int) getpid());
    return 0;
}
```

**Princeton University**  
**COS 217: Introduction to Programming Systems**  
**Trace of testforkret**

⊘

Copyright © 2018 by Robert M. Dondero, Jr.