

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

```
% gcc217 testforkwait.c -o testforkwait
```

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

```
% ./testforkwait
```

```
7275
```

```
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
            (int)getPid(), i);
        exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
            (int)getPid(), i);
    return 0;
}
```

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

```
% ./testforkwait
```

```
7275
```

```
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
            (int)getPid(), i);
        exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
            (int)getPid(), i);
    return 0;
}
```

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
            (int)getPid(), i);
        exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
            (int)getPid(), i);
    return 0;
}
```

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
              (int)getPid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
              (int)getPid(), i);
    return 0;
}
```

**Writes:**

7275 parent

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
            (int)getPid(), i);
        exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
            (int)getPid(), i);
    return 0;
}
```

**Writes:**

7275 parent

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

`% ./testforkwait`

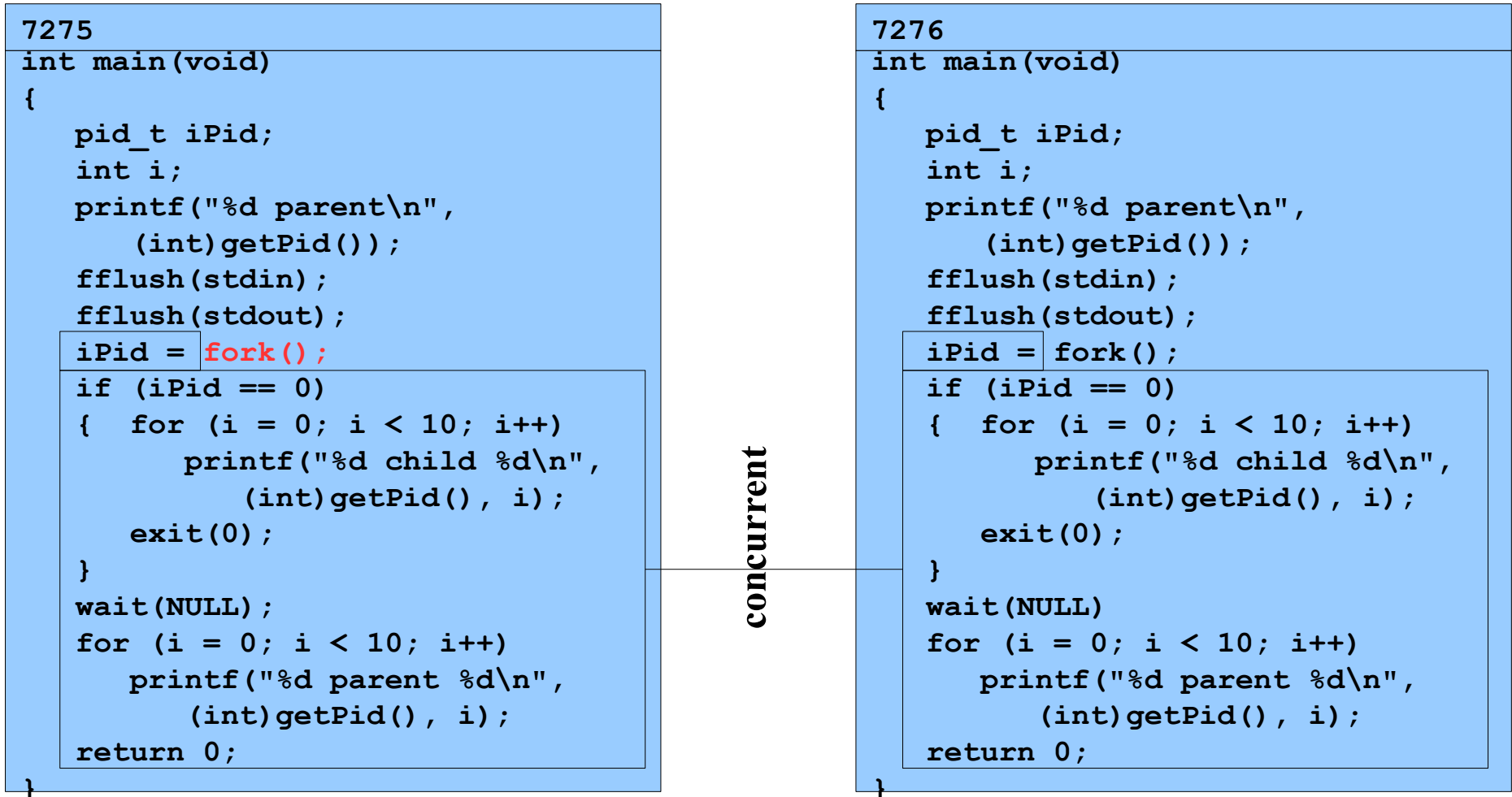
```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
            (int)getPid(), i);
        exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
            (int)getPid(), i);
    return 0;
}
```

**Writes:**

7275 parent

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

```
% ./testforkwait
```



Assume OS gives CPU to parent



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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

7276

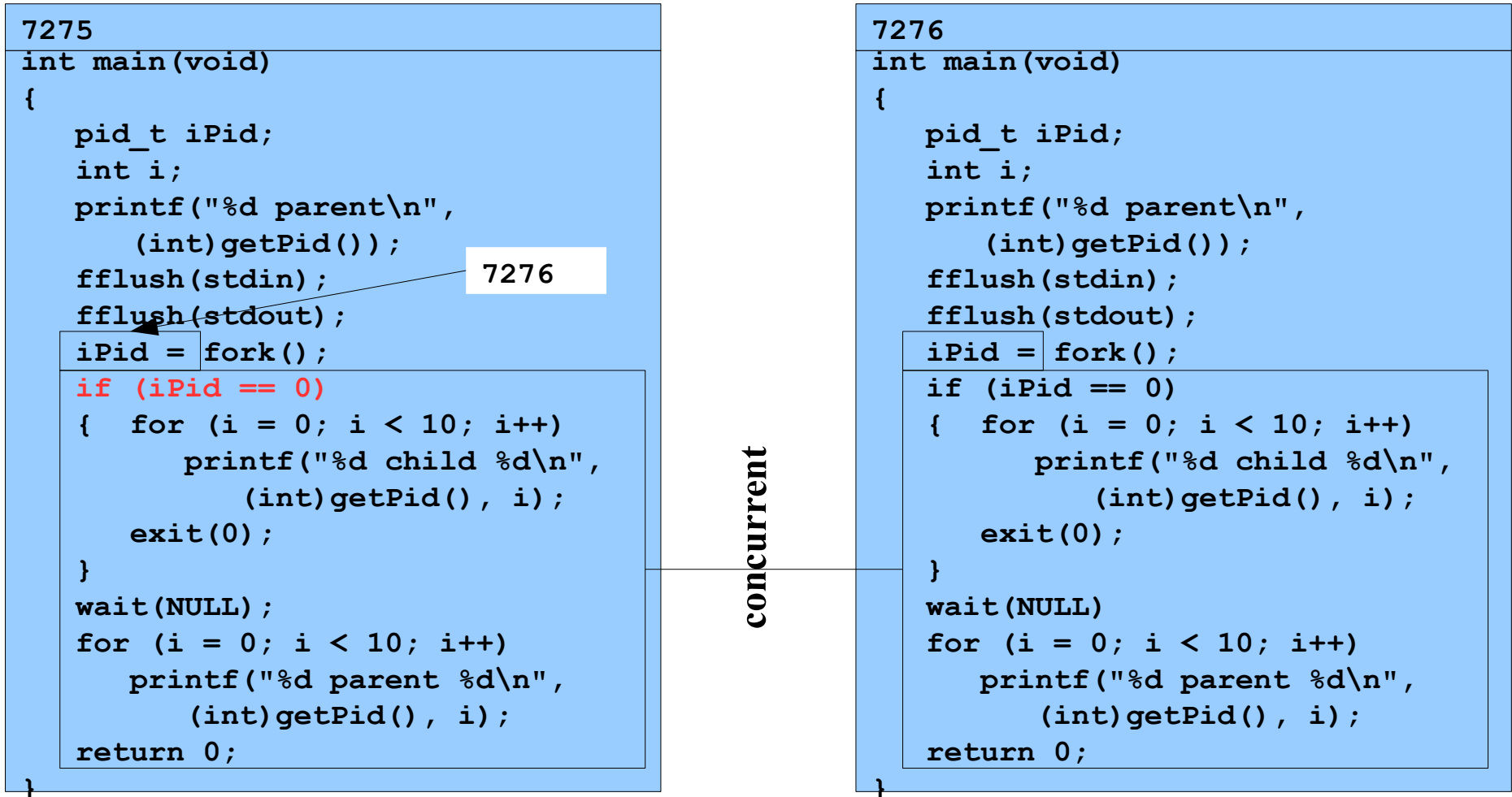
```
7276
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL)
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

concurrent

Assume OS gives CPU to parent

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

```
% ./testforkwait
```



Assume OS gives CPU to parent

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

7276

```
7276
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL)
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

concurrent

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

7276

```
7276
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL)
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

0

concurrent

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

7276

```
7276
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL)
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

0

concurrent

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

7276

```
7276
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL)
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

0

concurrent

Writes:  
7276 child 0 ... 9

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

% ./testforkwait

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
            (int)getPid(), i);
        exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
            (int)getPid(), i);
    return 0;
}
```

7276

```
7276
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
        (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
        printf("%d child %d\n",
            (int)getPid(), i);
        exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
            (int)getPid(), i);
    return 0;
}
```

0

concurrent

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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int)getPid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int)getPid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int)getPid(), i);
    return 0;
}
```

7276

Writes:

```
7275 parent 0 ... 9
```



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

```
% ./testforkwait
```

```
7275
int main(void)
{
    pid_t iPid;
    int i;
    printf("%d parent\n",
           (int) getpid());
    fflush(stdin);
    fflush(stdout);
    iPid = fork();
    if (iPid == 0)
    { for (i = 0; i < 10; i++)
      printf("%d child %d\n",
             (int) getpid(), i);
      exit(0);
    }
    wait(NULL);
    for (i = 0; i < 10; i++)
        printf("%d parent %d\n",
               (int) getpid(), i);
    return 0;
}
```

7276

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

8