
```

14  {
15      fprintf(stderr, "%s: %s\n", msg, gai_strerror(code));
16      exit(0);
17  }
18
19  void app_error(char *msg) /* Application error */
20  {
21      fprintf(stderr, "%s\n", msg);
22      exit(0);
23  }

```

code/src/csapp.c

图 A-1 (续)

A.2 错误处理包装函数

下面是一些不同错误处理包装函数的示例：

- Unix 风格的错误处理包装函数。图 A-2 展示了 Unix 风格的 `wait` 函数的包装函数。如果 `wait` 返回一个错误，包装函数打印一条消息，然后退出。否则，它向调用者返回一个 PID。图 A-3 展示了 Unix 风格的 `kill` 函数的包装函数。注意，这个函数和 `wait` 不同，成功时返回 `void`。

```

1  pid_t Wait(int *status)
2  {
3      pid_t pid;
4
5      if ((pid = wait(status)) < 0)
6          unix_error("Wait error");
7      return pid;
8  }

```

*code/src/csapp.c**code/src/csapp.c*图 A-2 Unix 风格的 `wait` 函数的包装函数

```

1  void Kill(pid_t pid, int signum)
2  {
3      int rc;
4
5      if ((rc = kill(pid, signum)) < 0)
6          unix_error("Kill error");
7  }

```

*code/src/csapp.c**code/src/csapp.c*图 A-3 Unix 风格的 `kill` 函数的包装函数

- Posix 风格的错误处理包装函数。图 A-4 展示了 Posix 风格的 `pthread_detach` 函数的包装函数。同大多数 Posix 风格的函数一样，它的错误返回码中不会包含有用的结果，所以成功时，包装函数返回 `void`。