

### 5. parse\_uri 函数

TINY 假设静态内容的主目录就是它的当前目录，而可执行文件的主目录是 `./cgi-bin`。任何包含字符串 `cgi-bin` 的 URI 都会被认为表示的是对动态内容的请求。默认的文件名是 `./home.html`。

图 11-33 中的 `parse_uri` 函数实现了这些策略。它将 URI 解析为一个文件名和一个可选的 CGI 参数字符串。如果请求的是静态内容(第 5 行)，我们将清除 CGI 参数字符串(第 6 行)，然后将 URI 转换为一个 Linux 相对路径名，例如 `./index.html`(第 7~8 行)。如果 URI 是用 `"/` 结尾的(第 9 行)，我们将把默认的文件名加在后面(第 10 行)。另一方面，如果请求的是动态内容(第 13 行)，我们就会抽取所有的 CGI 参数(第 14~20 行)，并将 URI 剩下的部分转换为一个 Linux 相对文件名(第 21~22 行)。

```

code/netp/tiny/tiny.c
1  int parse_uri(char *uri, char *filename, char *cgiargs)
2  {
3      char *ptr;
4
5      if (!strstr(uri, "cgi-bin")) { /* Static content */
6          strcpy(cgiargs, "");
7          strcpy(filename, ".");
8          strcat(filename, uri);
9          if (uri[strlen(uri)-1] == '/')
10             strcat(filename, "home.html");
11         return 1;
12     }
13     else { /* Dynamic content */
14         ptr = index(uri, '?');
15         if (ptr) {
16             strcpy(cgiargs, ptr+1);
17             *ptr = '\0';
18         }
19         else
20             strcpy(cgiargs, "");
21         strcpy(filename, ".");
22         strcat(filename, uri);
23         return 0;
24     }
25 }
code/netp/tiny/tiny.c

```

图 11-33 TINY `parse_uri` 解析一个 HTTP URI

### 6. serve\_static 函数

TINY 提供五种常见类型的静态内容：HTML 文件、无格式的文本文件，以及编码为 GIF、PNG 和 JPG 格式的图片。

图 11-34 中的 `serve_static` 函数发送一个 HTTP 响应，其主体包含一个本地文件的内容。首先，我们通过检查文件名的后缀来判断文件类型(第 7 行)，并且发送响应行和响应报头给客户端(第 8~13 行)。注意用一个空行终止报头。