```
1   char*
2   strcpy(char *dst, const char *src)
3   {
4     char *os;
5
6     os = dst;
7     while((*dst++ = *src++) != 0)
8       ;
9     return os;
10  }
11
12  int
13  strcmp(const char *p, const char *q)
14  {
15    while(*p && *p == *q)
16      p++, q++;
17    return (uchar)*p - (uchar)*q;
18  }
19
20  uint
21  strlen(const char *s)
22  {
23    int n;
24
25    for(n = 0; s[n]; n++)
26      ;
27    return n;
28  }
29
30  void*
31  memset(void *dst, int c, uint n)
32  {
33    stosb(dst, c, n);
34    return dst;
35  }
36
37  char*
38  strchr(const char *s, char c)
39  {
40    for(; *s; s++)
41      if(*s == c)
42        return (char*)s;
43    return 0;
44  }
45
46  char*
47  gets(char *buf, int max)
48  {
49    int i, cc;
50    char c;
51
```

```c
52    for(i=0; i+1 < max; ){
53      cc = read(0, &c, 1);
54      if(cc < 1)
55        break;
56      buf[i++] = c;
57      if(c == '\n' || c == '\r')
58        break;
59    }
60    buf[i] = '\0';
61    return buf;
62  }
63
64  int
65  stat(const char *n, struct stat *st)
66  {
67    int fd;
68    int r;
69
70    fd = open(n, O_RDONLY);
71    if(fd < 0)
72      return -1;
73    r = fstat(fd, st);
74    close(fd);
75    return r;
76  }
77
78  int
79  atoi(const char *s)
80  {
81    int n;
82
83    n = 0;
84    while('0' <= *s && *s <= '9')
85      n = n*10 + *s++ - '0';
86    return n;
87  }
88
89  void*
90  memmove(void *vdst, const void *vsrc, int n)
91  {
92    char *dst;
93    const char *src;
94
95    dst = vdst;
96    src = vsrc;
97    while(n-- > 0)
98      *dst++ = *src++;
99    return vdst;
100 }
```