

## 字符串读入 - string

```
while(getline(cin, s))
{
    if(s == "") continue;
    prg[i].push_back(s);
    if(prg[i].back() == "end") break;
}
```

```
string s;
getline(cin, s);
```

注意点：

- \* 首行数字后有一个换行符需要读入
- \* 防止大量行末空格，C要把缓冲区开大一些，C++可用string
- \* 注意空行（空行不能简单的用length==0判断，有可能有一些空格在里面的空行）
- \* 注意文件尾：如果用fgets，需要判断是否有\n，也有可能没有（文件尾）
- \* 如果要用cin istream，使用ios::sync\_with\_stdio(false); 并且全部数据使用cin读入。速度堪比scanf

ps. C的字符串处理真是弱啊。如果用fgets，必须引入“每一行的字符串长度有上限”假设。如果每行字符串后面有大量空格。。就坑了。最好的办法是用getchar逐字符读入直到EOF。

一段C-style的处理

```
while(fgets(prg[i].s[j], MAX_LINE, stdin)){
    int len = strlen(prg[i].s[j]);
    if(prg[i].s[j][len-1] == '\n'){
        prg[i].s[j][len-1] = 0;
        len -= 1;
    }
    if(len > 0){
        j += 1;
        if(prg[i].s[j-1][2] == 'd'){
            break;
        }
    }
}
```

一段C++处理（用length >= 3判断是因为题目原因，防止有1个空格的空行出现）

```
while(getline(cin, prg[i].code[j++])){
    if(prg[i].code[j-1].length() >= 3){
        if(prg[i].code[j-1][2] == 'd'){
            break;
        }
    }
    else{
        j -= 1;
    }
}
```