功能类似于windwos中的批处理，就是把经常要干，比较复杂的，命令很多的的工作，写成shell脚本，以后直接执行这个脚本或者让机器自动运行这个脚本。

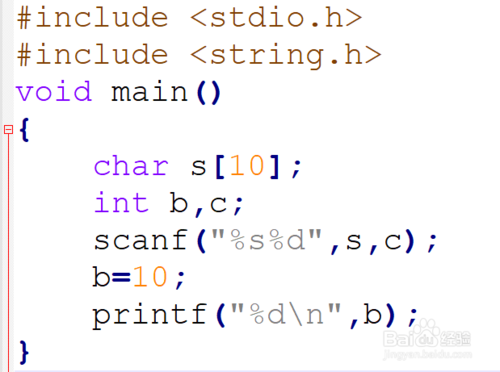
不需要使用编译器编译即可直接运行。比如说启动一些需要大量手工操作的程序，将执行过程写成脚本，每次启动的时候运行脚本方便快捷

shell脚本是shell的一种扩展功能，也就是shell会根据你所写的脚本一条一条的去执行，C写的程序是直接给CPU执行的。一般来说做一些简单功能，特别是和shell有关的。我们会用shell脚本，如系统环境的设定，开机后执行系统的初始化等。做一些功能复杂的，特别人用户自已所想到的功能时会用C语。另外如果程序执行是要用到root权限是最好也用C语。

（1）Shell脚本与Windows/Dos下的批处理相似，也就是用各类命令预先放入到一个文件中，方便一次性执行的一个程序文件，主要是方便管理员进行设置或者管理用的。但是它比Windows下的批处理更强大，比用其他编程程序编辑的程序效率更高，它使用了Linux/Unix下的命令。

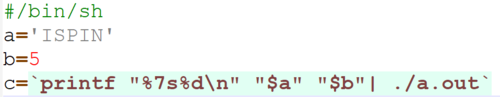
Shell 里可以调用C程序。在shell中调用自己编写的c程序，程序需要手动输入数据，需要将shell脚本中的变量作为输入数据给c程序。并把c程序的输出赋值给shell的变量。

1. 以下图中简单的c程序作为范例



1. 2

示意脚本如下



C 程序可以调用脚本

Shell应用场景

Shell能做什么

将一些复杂的命令简单化(平时我们提交一次github代码可能需要很多步骤，但是可以用Shell简化成一步)

可以写一些脚本自动实现一个工程中自动更换最新的sdk(库)

自动打包、编译、发布等功能

清理磁盘中空文件夹

总之一切有规律的活脚本都可以尝试一下

Shell不能做什么

需要精密的运算的时候

需要语言效率很高的时候

需要一些网络操作的时候

总之Shell就是可以快速开发一个脚本简化开发流程，并不可以用来替代高级语言

Shell的工作原理

Shell可以被称作是脚本语言，因为它本身是不需要编译的，而是通过解释器解释之后再编译执行，和传统语言相比多了解释的过程所以效率会略差于传统的直接编译的语言。

最简单的脚本:

#!/bin/bash

echo "Hello World"

只需要打开文本编辑工具，编辑成以上的样子,然后保存成test.sh