一.有以下代码片段：D

runlevel=1

for s in /etc/rc$runlevel.d/K\*

do

level=${s#/etc/rc$runlevel.d/K} 删除左边的/etc/rc$runlevel.d/K

level=${level%%[a-zA-Z]\*} 删除右边的字母

done

则以下说法错误的是：

1. 循坏时,s将依次取/etc/rc1.d/下所有以K开头的名字
2. 式子中的${s#/etc/rc$runlevel.d/K}表示去除变量s左边的一部分字符
3. 式子中的${level%%[a-zA-Z]\*}表示去除变量s右边的一部分字符
4. level最终值是runlevel
5. 以下说法错误的是：D
6. test $n1 -eq $n2 用于比较两个数值n1和n2是否相等
7. test $s1 = $s2 用于比较两个字符串s1和s2是否相等
8. test -e “file” 用于判断文件file是否存在
9. test -e $v 用于判断变量v是否已经定义
10. 以下代码有语法错误的是：C
11. if [ -e $file ];

then

echo “ok”

fi

1. if test -e $file

then

echo “ok”

else

echo “no”

fi

1. if [ -e $file ]

then

echo “ok”

;; -> 只有case分支才有;;

else

echo “no”

;;

fi

D.if [ -e $file1 ]

then

echo “ok1”

elif [ -e $file2 ]

then

echo “ok2”

fi

1. 以下说法错误的是：E
2. SHELL脚本中可以直接对变量赋值，例如age=12。
3. SHELL脚本中可以使用read命令来接收用户的输入，例如read age。
4. SHELL脚本中可以使用命令行传递参数，例如age=$1。
5. SHELL脚本中可以继承环境变量，例如PATH。
6. SHELL脚本中无法读取文件，只能读写变量。可以使用cat来读取文件
7. 判断题

SHELL脚本中，某一条语句执行出错了，只要不是语法错误，下面的语句可以继续运行。

可以

1. 将以下C语言程序效果改写为SHELL脚本

#include <stdio.h>

int main()

{

int n;

while(1)

{

printf("================================\n");

printf(" 1. one \n");

printf(" 2. two \n");

printf(" 3. three \n");

printf(" 4. four \n");

printf("================================\n");

printf("please input a num:");

scanf("%d",&n);

switch(n)

{

case 1:

printf("one!\n");

break;

case 2:

printf("two!\n");

break;

case 3:

printf("three!\n");

break;

case 4:

printf("four!\n");

break;

default:

printf("unknown!\n");

break;

}

}

return 0;

}

Shell:

#!/bin/bash

declare -i n=0

while [ $n -eq 0 ]

do

echo "=========================="

echo " 1. one "

echo " 2. two "

echo " 3. three "

echo " 4. four "

echo "=========================="

echo "pls input your num"

read a

case $a in

1) echo "one";;

2) echo "two";;

3) echo "three";;

4) echo "four";;

\*) echo "unknown";;

esac

done