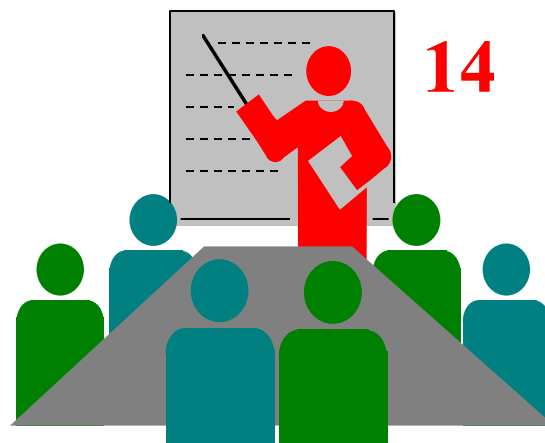


第五章 数据有格式输入输出

1. 概述
2. 格式说明与格式编辑符
3. 整数有格式输入输出
4. 实数有格式输入输出
5. 复数有格式输入输出
6. 字符串有格式输入输出
7. 逻辑值有格式输入输出
8. 二、八、十六进制数据有格式输入输出
9. 任意类型数据有格式输入输出
10. 不可重复编辑符
11. 纵向走纸控制
12. 输入输出表与格式说明表相互关系
13. 有格式输入输出应用举例
14. 习题五



5.1 概述

- 输入输出数据是否简洁、直观、醒目、规范,是反映程序质量的一项重要指标。
- 示例:某程序需输入3个学生4门课成绩数据,共输入12个数据,有两种输入方式:

① 78.5 85 58 98 75 88.5 99 45.5 84.5 69.5 77 82 ✓

② 数学 物理 化学 英语

输入第1位同学四门课成绩: 78.5 85.0 58.0 98.0 ✓

输入第2位同学四门课成绩: 75.0 88.5 99.0 45.5 ✓

输入第3位同学四门课成绩: 84.5 69.5 77.0 82.0 ✓

- 示例:某程序要输出3个学生4门课成绩数据,共输出12个数据,有两种输出方式:

① 78.50000 85.00000 58.00000 98.00000 75.00000
88.50000 99.00000 45.50000 84.50000 69.50000
77.00000 82.00000

② 数学 物理 化学 英语

第1位同学成绩: 78.5 85.0 58.0 98.0

第2位同学成绩: 75.0 88.5 99.0 45.5

第3位同学成绩: 84.5 69.5 77.0 82.0

输入分类
输入提示

简洁直观
醒目规范

5.2 格式说明与格式编辑符



- ◆ 示例
- ◆ 格式说明
- ◆ 编辑符



```
① READ 100 m,n
③ READ(*,100)m,n
  WRITE(*,200)m+n,m-n
  100 FORMAT(I3,I3)
  200 FORMAT(1X, 'm+n=',I4, 'm-n=',I4)
  END
④ READ(*,"(I3,I3)")m,n
  WRITE(*,"(1X, 'm+n=',I4, 'm-n=',I4)")m+n,m-n
  END
```



5.2 格式说明与格式编辑符



- ◆ 示例
- ◆ 格式说明
- ◆ 编辑符



- 数据有格式输入输出需要给出相关的格式说明。
- **FORMAT**语句提供输入输出数据的格式说明信息。
- **FORMAT**语句既不是执行语句,也不是说明语句。
- **FORMAT**语句可出现在说明语句之后,**END**语句之前任何位置。
- 格式说明表是关键字**FORMAT**之后的括号部分。
- 格式说明表由若干格式编辑符组成,编辑符由“,”分隔。
- 格式说明表可嵌入到输入输出语句中,取代**FORMAT**语句。
- 格式说明一般使用**FORMAT**语句,简单格式使用嵌入方式。
- **FORMAT**语句语法描述:
 - <**FORMAT**语句> → <语句标号> **FORMAT** <格式说明表>
 - <格式说明表> → (<格式说明>{[,|/]<格式说明>})



5.2 格式说明与格式编辑符



- ◆ 示例
- ◆ 格式说明
- ◆ 编辑符

- FORTRAN 90提供了24个格式编辑符, 如: I、F、E、A等。
 - 格式编辑符分为两类: 可重复编辑符、不可重复编辑符。
 - 可重复编辑符可指定重复系数, 与I/O表中的一个I/O项对应。
 - 不可重复编辑符不可指定重复系数, 不与输入输出项相对应。
- 如: I、F、E、A等编辑符为可重复编辑符。
- 如: X、/、\等编辑符为不可重复编辑符。

```
READ 100, num1, num2, price1, price2
100 FORMAT(2I3, F7.1, E8.2)
PRINT 200, num1*price1+num2*price2
200 FORMAT(1X,'total prices=',E14.7)
END
```

输入: 125□2195234.57.52E+04✓。

输出: total prices=□1.3483512E+07。



表5-1 可重复编辑符

格式	名称	功能	输入	输出		
[r]Im[.n]	整型编辑符	将整数按指定域宽输入输出	✓	✓		
[r]Bm[.n]	二进制编辑符	将2进制数按指定域宽输入输出	✓	✓		
[r]Om[.n]	八进制编辑符	将8进制数按指定域宽输入输出	✓	✓		
[r]Zm[.n]	16进制编辑符	将16进制数按指定域宽输入输出	✓	✓		
[r]Fm.d	小数型实型编辑符	将实数按指定域宽输入输出	✓	✓		
[r]Em.d[Ee]	指数型实型编辑符	将实数按指定域宽输入输出	✓	✓		
[r]Dm.d	双精度实型编辑符	将双精度数按指定域宽输入输出	✓	✓		
[r]ENm.d[Ee]	工程计数法编辑符	将实数按工程计数法输入输出	✓	✓		
[r]ESm.d[Ee]	科学计数法编辑符	将实数按科学计数法输入输出	✓	✓		
[r]A[m]	字符型编辑符	将字符串按指定域宽输入输出	✓	✓		
[r]Lm	逻辑型编辑符	将逻辑值按指定域宽输入输出	✓	✓		
[r]Gm.d[Ee]	通用编辑符	将任意类型数据按域宽输入输出	✓	✓		

表5-2 不可重复编辑符

格式	名称	功能	输入	输出		
字符串	字符串编辑符	将字符串输出		✓		
nH	H编辑符	将H后n个字符输出		✓		
Q	字符计数编辑符	获到输入记录中剩余字符数		✓		
Tc, TLc, TRC	位置编辑符	指定输入输出记录中的位置		✓	✓	
nX	位置编辑符	指定输入输出记录中的位置		✓	✓	
SP, SS, S	+号编辑符	控制+号输出		✓		
/	斜杠编辑符	终止本记录,开始下一记录		✓	✓	
\	反斜杠编辑符	下一记录接上一记录输出			✓	
\$	\$编辑符	下一记录接上一记录输出		✓		
:	格式控制编辑符	无输出数据终止格式控制			✓	
KP	比例因子编辑符	设置F和E编辑符指数比例因子		✓	✓	
BN, BZ	空格替代编辑符	指定数字串中空格的含义		✓		

5.3 整数有格式输入输出



- ◆ 格式 →
- ◆ 输入
- ◆ 输出

| 编辑符可用于整数的有格式输入输出,一般格式是:

[r]lw[.m]

其中: r为l编辑符重复次数。

即: [r]lw[.m]等价于lw[.m],lw[.m],...,lw[.m]。

w为整数输入输出域宽,即整数占连续字符个数。

m为输出最少位数。

[.m]项只对整数输出有效,对输入无效。

程序示例:

```
INTEGER(2) i,j,k
READ 10,i,j,k
10  FORMAT(I4,2I5)    ! 等价于FORMAT(I4,I5,I5)
PRINT "(1X, 'i+j+k=',I5)",i+j+k
END
```



5.3 整数有格式输入输出



- ◆ 格式
- ◆ 输入
- ◆ 输出

- 输入数据间可按规定域宽紧密输入,也可用逗号间隔输入,不用空格。
- 在域宽内,数字前后,数字之间可出现空格,这些空格一般被视为空。
- 在域宽内,全为空格,输入整数为0。
- 在域宽内,输入数据不能超出范围,不能是非十进制整数。
- I编辑符要求输入表中对应变量的类型一般是整型,否则会产生错误。
- 示例:

```
INTEGER(2) i,j,k
READ 10,i,j,k
10  FORMAT(I4,2I5)  ! 等价于FORMAT(I4,I5,I5)
PRINT "(1X, 'i+j+k=',I5)",i+j+k
END
```



5.3 整数有格式输入输出



- ◆ 格式
- ◆ 输入
- ◆ 输出

- 若域宽等于整数位数,则在域宽内输出该整数,与后一数据无空格间隔。
- 若域宽大于整数位数,则在域宽内按右对齐输出该整数,前补空格。
- 若域宽小于整数位数,则表明域宽不足,无法输出数据,域宽内填充“*”。
- 对于lw.m格式,m表示输出最小位数。位数小于m,补前导0至m位。
- 示例:

```
READ *,i,j,k
10  FORMAT(1X, 'i=',I4, 'j=',I4, 'k=',I4)
PRINT 10,i,j,k
END
```

输入:

- 1234,-
- 123,4567✓
- 10,-12,1234✓
- 12345,-
- 12,1234✓
- 8,-12,1234✓

输出:

- i=1234j=-
- 123k=5678
- i=□□10j=□-
- 12k=1234
- i=****j=□-
- 12k=1234
- i=□008j=-
- 012k=1234



5.3 整数有格式输入输出

输入数据:

◆ 格式
◆ 输入
◆ 输出

● 123412345-

6789✓

● 12,12345-6789✓

● 12,34,□56789✓

● 1234,□56789✓

● □1□2□123□-

□78□✓

● 1□,12□34□-

6789✓

● 1□,12□345-

6789✓

● □1□2□□□□-

□78□✓

● 123454321-

6789✓

● 2#1112345-

6789✓

● 1234543.2-

6789✓

i、j、k 值:

● i=1234,j=12345,k=-6789

● i=12,j=12345,k=-6789

● i=12,j=34,k=5678

● i=1234,j=0,k=5678

● i=12,j=123,k=-78

● i=1,j=1234,k=-678

● 产生数据转换错误

● i=12,j=0,k=-78

● 产生数据转换错误,超出最大范围

● 产生数据转换错误,非十进制数。

● 产生数据转换错误,数据为实数。



5.4 实数有格式输入输出



- ◆ 格式F →
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

F 编辑符可用于小数型实数的有格式输入输出,一般格式是:

[r]Fw.d

其中: r为F编辑符重复次数。

即: **[r]Fw.d** 等价于 **Fw.d,Fw.d,...,Fw.d**。

w为小数型实数输入输出域宽,即实数占连续字符个数。

d为小数位数。对于输入,无小数点,小数点由**d**确定,有小数点,小数点由域宽内实际位置确定。对于输出,输出数据域宽包括一个小数点。

程序示例:

```
READ 10,a, b, c
10  FORMAT(2F5.2,F8.3)
PRINT "(1X,'a+b+c=',F8.3)",a+b+c
END
```



5.4 实数有格式输入输出



- ◆ 格式F
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

- 若输入数据不带小数点,则按域宽读取。并根据d值自动加小数点。
- 若输入数据自带小数点,则按域宽读取。自带小数点优先d值。
- 输入数据可紧密输入,也可逗号间隔。逗号有强行终止域宽作用。
- 在规定域宽内,数字前后,数字之间可出现空格,一般空格被视为空。
- 在规定域宽内,全为空格,输入实数为0.0。
- 在规定域宽内,可按指数形式输入数据,小数部分小数点d值确定。
- 在规定域宽内,输入数据不能超出取值范围。
- 对于F编辑符,输入表中对应变量类型必须是实型或复型。
- 示例:

```
READ 10,a, b, c
10  FORMAT(2F5.2,F8.3)
PRINT "(1X,'a+b+c=',F8.3)",a+b+c
END
```



5.4 实数有格式输入输出



- ◆ 格式F
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

- 对正数,若 $w=t+d+1$,对负数,若 $w=t+d+2$,且 $d=s$,则在域宽内输出该实数。
- 对正数,若 $w=t+d+1$,对负数,若 $w=t+d+2$,且 $d>s$,则在域宽内左对齐输出该实数,小数部分后续补 $d-s$ 个0。
- 对正数,若 $w=t+d+1$,对负数,若 $w=t+d+2$,且 $d<s$,则在域宽内左对齐输出该实数,小数部分保留 d 位,多余部分按“四舍五入”处理。
- 对正数,若 $w>t+d+1$,对负数,若 $w>t+d+2$,则在域宽内右对齐输出该实数,整数前补 $w-(t+d+1)$ 或 $w-(t+d+2)$ 个空格,小数部分按前面3种情况处理,占 d 位。
- 对正数,若 $w<t+d+1$,对负数,若 $w<t+d+2$,则域宽不足,在域宽内填充“*”。
- 示例:

```
READ *,a,b
WRITE(*,10) a,b
10 FORMAT(1X,'a=',F7.3,'b=',F7.3)
END
```

设：输出数据的整数部分位数为 t ,小数部分位数为 s 。



5.4 实数有格式输入输出



- ◆ 格式F
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

E 编辑符可用于指数型实数的有格式输入输出,一般格式是:
[r]Ew.d[Ee]

其中: r为F编辑符重复次数。

即: **[r]Ew.d[Ee]** 等价于 **Ew.d[Ee],Ew.d[Ee],...,Ew.d[Ee]**。

w为指数型实数输入输出域宽,即实数占连续字符个数。

d为小数位数,其含义同F编辑符。

e为指数位数,缺省为2,不包括E和符号,指数域宽**e+2**。

程序示例:

```
READ 11,a, b, c
11 FORMAT(E5.2,E8.3,E8.3E3)
PRINT "(1X,'a+b+c=',E12.3)",a+b+c
END
```



5.4 实数有格式输入输出



- ◆ 格式F
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

- 对于输入,E编辑符功能类似F编辑符。
- 在规定域宽内,可按小数型实数输入,也可按指数型实数输入。
- 示例:

```
READ 11,a, b, c
11 FORMAT(E5.2,E8.3,E8.3E3)
PRINT "(1X,'a+b+c=',E12.3)",a+b+c
END
```

输入:

● 123454321E□+2123E□+
12✓

a,b,c 值:

● a=123.45, b=432.1,
c=1.23×10¹¹



5.4 实数有格式输入输出



- ◆ 格式F
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

- 对正数,若 $w=d+5$,对负数,若 $w=d+6$,且 $d=t$,则在域宽内输出该实数。
- 对正数,若 $w=d+5$,对负数,若 $w=d+6$,且 $d>t$,则在域宽内左对齐输出该实数,小数部分后续补 $d-t$ 个0。
- 对正数,若 $w=d+5$,对负数,若 $w=d+6$,且 $d<t$,则在域宽内左对齐输出该实数,小数部分保留 d 位,多余部分按“四舍五入”处理。
- 对正数,若 $w>d+5$,对负数,若 $w>d+6$,则在域宽内右对齐输出该实数,整数前补 $w-(d+5)-1$ 或 $w-(d+6)-1$ 个空格,小数部分按前面3种情况处理。
- 对正数,若 $w<d+5$,对负数,若 $w<d+6$,则域宽不足,在域宽内填充“*”。
- 对双精度实数,指数有3位,指数部分的E字母省略,保持3位指数。
- 示例:

```
READ *,a,b
WRITE(*,10) a,b
10 FORMAT(1X,'a=',E10.5,'b=',E11.5)
END
```

设: 输出实数的有效位数为 t 。



5.4 实数有格式输入输出



- ◆ 格式F
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

输入:

a,b,c 值:

● 1234554321-1234567✓

● a=123.45,b=543.21,c=-

● 1.23454321-1.234567✓

1234.567

● 12345432.1-1234567✓

● a=1.234,b=543.21,c=-

● 123,1.23,□□-□6789✓

1.23456

● 1□2□35□432□1□23□45□6

● a=123.45,b=432.1,c=-

7✓

1234.567

● □1□2354□32□-

● a=1.23,b=1.23,c=-6.789

□123□456□✓

● a=1.23,b=54.32,c=12.345

● 1□23,□543212345678✓

● a=1.23,b=54.32,c=-1.234

● 12345□□□□-

● a=1.23,b=54.32,c=12345.6

□□12345678□✓

8

● 12345543211234E+02✓

● a=123.45,b=0.0,c=-12.345

● 12345543211234E2✓

● a=123.45,b=543.21,c=123.

● 12345543211234+2✓

4

● 12345543211234E+40✓

● a=123.45,b=543.21,c=123.

● 12345543211234E-50✓

4

● a=123.45,b=543.21,c=123.

4

● 产生数据转换错误(“上溢” 错

● a=123.45, b=543.21, c=0.0



5.4 实数有格式输入输出



- ◆ 格式F
- ◆ 输入F
- ◆ 输出F
- ◆ 格式E
- ◆ 输入E
- ◆ 输出E

- 对正数,若 $w=t+d+1$,对负数,若 $w=t+d+2$,且 $d=s$,则在域宽内输出该实数。
- 对正数,若 $w=t+d+1$,对负数,若 $w=t+d+2$,且 $d>s$,则在域宽内左对齐输出该实数,小数部分后续补 $d-s$ 个0。
- 对正数,若 $w=t+d+1$,对负数,若 $w=t+d+2$,且 $d<s$,则在域宽内左对齐输出该实数,小数部分保留 d 位,多余部分按“四舍五入”处理。
- 对正数,若 $w>t+d+1$,对负数,若 $w>t+d+2$,则在域宽内右对齐输出该实数,整数前补 $w-(t+d+1)$ 或 $w-(t+d+2)$ 个空格,小数部分按前面3种情况处理,占 d 位。
- 对正数,若 $w<t+d+1$,对负数,若 $w<t+d+2$,则域宽不足,在域宽内填充“*”。

● 示例:

输入:

- 125.511,-12.233✓
- 125.5,-12.23✓
- 125.5234,-12.233567✓
- 1.234,-2.233567✓
- 1111.23,-112.23✓

输出:

- a=125.511b=-12.233
- a=125.500b=-12.230
- a=125.523b=-12.234
- a=□□1.234b=□-2.234



5.4 实数有格式输入输出



◆ 格式F
◆ 输入F
◆ 输出F
◆ 格式E
◆ 输入E
◆ 输出E

- 对正数,若 $w=d+5$,对负数,若 $w=d+6$,且 $d=t$,则在域宽内输出该实数。
- 对正数,若 $w=d+5$,对负数,若 $w=d+6$,且 $d>t$,则在域宽内左对齐输出该实数,小数部分后续补 $d-t$ 个0。
- 对正数,若 $w=d+5$,对负数,若 $w=d+6$,且 $d<t$,则在域宽内左对齐输出该实数,小数部分保留 d 位,多余部分按“四舍五入”处理。
- 对正数,若 $w>d+5$,对负数,若 $w>d+6$,则在域宽内右对齐输出该实数,整数前补 $w-(d+5)-1$ 或 $w-(d+6)-1$ 个空格,小数部分按前面3种情况处理。
- 对正数,若 $w<d+5$,对负数,若 $w<d+6$,则域宽不足,在域宽内填充“*”。
- 对双精度实数,指数有3位,指数部分的E字母省略,保持3位指数。
- 示例:

```
READ *,a,b
WRITE(*,10) a,b
10 FORMAT(1X,'a=',E10.5,'b=',E11.5)
```

输入:

● 12.543,-122.33✓
● 12.5,-122.3✓
● 12.53456,-
122.3344✓

输出:

● a=.125 43E+02b=-
.12233E+03
● a=.125 00E+02b=-
.12230E+03
● a=.125 35E+02b=-
.12233E+03



5.5 复数有格式输入输出



◆ 概述 →

复数编辑符是通过两个实型编辑符实现的,第一个对应实部,第二个对应虚部。

程序示例:

```
COMPLEX c1,c2
READ 23,c1,c2
23  FORMAT(F5.2,E5.2,F8.3,E8.3)
PRINT "(1X,'c1+c2=', F7.2,E10.3)",c1+c2
END
```

输入: 12345123E24321.5,1234E-1✓。

c1和c2值: $c1=(123.45,123.0)$, $c2=(4321.5,1.234 \times 10^{-1})$ 。

输出: $c1+c2=4444.95 \square\square 0.123E+03$ 。

复数输出首尾无括号,中间无逗号,两数之间用1个空格间隔。



5.6 字符串有格式输入输出



- ◆ 格式 →
- ◆ 输入
- ◆ 输出

A 编辑符可用于字符串的有格式输入输出,一般格式是:

[r]A[w]

其中: r为A编辑符重复次数。

即: [r]A[w]等价于A[w],A[w],...,A[w]。

w为字符串输入输出域宽,即字符串占连续字符个数。

缺省w,域宽为输入和输出表中对应字符型变量的长度。

程序示例:

```
CHARACTER str1*5,str2*4,str3*3,str4*3
READ 24,str1,str2,str3,str4
24   FORMAT(A5,A4,2A3)
PRINT *,str1//str2//str3//str4
END
```



5.6 字符串有格式输入输出



◆ 格式 ◆ 输入 ◆ 输出

- 域宽等于变量长度,按域宽依次截取字符串,赋予变量,不足补空格。
- 域宽小于变量长度,按域宽依次截取字符串,补足空格后赋于变量。
- 域宽大于变量长度,按域宽依次截取字符串,取后部字符赋于变量。
- 缺省域宽,域宽默认取输入表对应字符型变量长度。
- 输入字符串不能加引号。引号、逗号和空格按一般字符对待。
- 示例:

```
CHARACTER s1*5,s2*4,s3*3,s4*3  
READ 24,s1,s2,s3,s4  
24   FORMAT(A5,A4,2A3)  
PRINT *,s1//s2//s3//s4  
END
```



5.6 字符串有格式输入输出



- ◆ 格式
- ◆ 输入
- ◆ 输出



- 域宽等于表达式值长度,则按域宽输出字符串,字符串中空格有效。
语句: `PRINT "(1X,'#',A15,'#')", 'ABC□□'/'DEFG'/'□HI'/'J□K'`
输出: `#ABC□□DEFG□HI□K#`
- 域宽小于表达式值长度,则按域宽截取左侧子字符串输出。
语句: `PRINT "(1X,'#',A10,'#')", 'ABC□□'/'DEFG'/'□HI'/'J□K'`
输出: `#ABC□□DEFG□#`
- 域宽大于表达式值长度,则将表达式值左补空格至域宽输出。
语句: `PRINT "(1X,'#',A18,'#')", 'ABC□□'/'DEFG'/'□HI'/'J□K'`
输出: `#□□□ABC□□DEFG□HI□K#`
- 缺省域宽,以表达式值长度为域宽输出。
语句: `PRINT "(1X,'#',A,'#')", 'ABC□□'/'DEFG'/'□HI'/'J□K'`
输出: `#ABC□□DEFG□HI□K#`



5.6 字符串有格式输入输出

◆ 格式 ◆ 输入 ◆ 输出

- 域宽等于变量长度,按域宽依次截取字符串,赋予变量,不足补空格。
- 域宽小于变量长度,按域宽依次截取字符串,补足空格后赋予变量。
- 域宽大于变量长度,按域宽依次截取字符串,取后部字符赋予变量。

输入:

s1,s2,s3 值:

- ABCDEFGHIJKLMNO ✓ ● s1='ABCDE',s2='FGHI',s3='JKL',s4='MNO'
- ABCDEFGHIJKLM ✓ → ● s1='ABCDE',s2='FGHI',s3='JKL',s4='M '
- ABCDEFGHIJKLMNOPQRS ✓ ● s1='ABCDE',s2='FGHI',s3='JKL',s4='MNO'
- FORMAT改写成FORMAT(4A4)
● ABCDEFGHIJKLMNOPQRST ≠ 'ABCD□',s2='EFGH',s3='JKL',s4='NOP'
- FORMAT改写成FORMAT(4A)
● ABCDEFGHIJKLMNOPQRST ≠ 'ABCDE',s2='FGHI',s3='JKL',s4='MNO'
- 'china','xian','new','pen' ✓
● s1=" 'chin",s2="a",' ",s3="xia",s4="n',"



5.7 逻辑值有格式输入输出



- ◆ 格式 →
- ◆ 输入
- ◆ 输出

L 编辑符可用于逻辑值的有格式输入输出,一般格式是:

[r]Lw

其中: r为L编辑符重复次数。

即: [r]Lw等价于Lw,Lw,...,Lw。

w为逻辑值输入输出域宽,即逻辑值占连续字符个数。

缺省w,域宽为2。

程序示例:

```
LOGICAL log1,log2,log3,log4
READ  26,log1,log2,log3,log4
PRINT 27, log1,log2,log3,log4
26    FORMAT(L2,L1,L2,L3)
27    FORMAT(1X,L2,L1,L2,L3)
END
```



5.7 逻辑值有格式输入输出



◆ 格式
◆ 输入
◆ 输出

- 域宽内首个非空格或非小数点“.”字符是“T”或“t”,则输入为.TRUE.。
- 域宽内首个非空格或非小数点“.”字符是“F”或“f”,则输入为.FALSE.。
- 在域宽内,字母T、t、F、f后可跟任意字符,不影响逻辑值输入。
- 逻辑值间可用逗号分隔,不用空格。
- 输入逻辑值.TRUE.,域宽必须大于5。输入.FALSE.,域宽必须大于6。
- 示例:

```
LOGICAL log1,log2,log3,log4
READ 26,log1,log2,log3,log4
PRINT 27, log1,log2,log3,log4
26   FORMAT(L2,L1,L2,L3)
27   FORMAT(1X,L2,L1,L2,L3)
END
```

输入: ☐TFTA☐FB✓

值: log1 = .TRUE.,log2 = .FALSE.,log3 = .TRUE.,log4 = .FALSE.



5.7 逻辑值有格式输入输出



- ◆ 格式
- ◆ 输入
- ◆ 输出



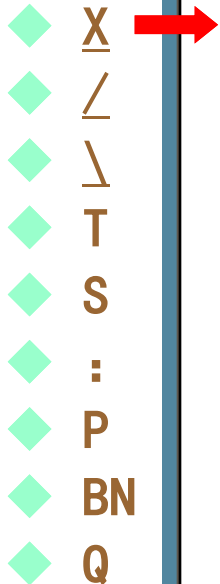
- 在指定域宽内右对齐输出T(.TRUE.)或F(.FALSE.)。
- 示例: LOGICAL log1,log2,log3,log4

```
Log1=.TRUE.;log2=.TRUE.;log3=.FALSE.;log4=.FALSE.  
PRINT 27,log1,log2,log3,log4  
27    FORMAT(1X,L2,L1,L2,L3)  
END
```

输出: □TT□F□□F



5.10 不可重复编辑符



- **X**编辑符用于设置输入输出数据的开始位置。
- 对输出,从当前位置开始插入若干空格,下一输出数据从新位置开始进行。
- 对输入,从当前位置开始右移若干字符,下一输入数据从新位置开始进行。
- 示例:

```
      READ 11,i,j,k
11  FORMAT(2X,I3,3X,I3,3X,I3)
      PRINT 21,i,j,k,i+j+k
21  FORMAT(1X,'i=',I3,4X,'j=',I3,4X,'k=',I3,4X,'i+j+k=',I4)
      END
```

输入: i=123□j=456□k=789✓。

i、j、k值: i=123,j=456,k=789。

输出: □i=123□□□□j=456□□□□k=789□□□□i+j+k=1368。



5.10 不可重复编辑符



- /编辑符用于输入输出数据时结束本行操作,后面数据从下一行开始。
- 其作用是用一个输入输出语句完成多行输入输出操作。
- /编辑符可替代“,”号作为说明表中编辑符间的间隔符。

● 示例: **READ 10,i,a,j,b**

PRINT 20,i,a,j,b

10 FORMAT(I5,3X,F5.1 /I5,3X,F5.1)

20 FORMAT(1X,I5,3X,F5.1 /1X,I5,3X,F5.1)

END

输入:

□□123.4□□345□□□567.8✓

□4321□□□432.1✓

输出:

□□123□□□34.5✓

□4321□□□432.1✓

◆ X
◆ /
◆ \
◆ T
◆ S
◆ :
◆ P
◆ BN
◆ Q



/ 编辑符

1) / 编辑符用于数据输入

- ◆ 数据分多行输入要与输入格式说明表中斜杠编辑符输入输出项一致，否则丢失部分数据。
- ◆ 在输入格式说明表中可使用连续多个斜杠编辑符输入，但要注意对输入数据的正确分行。
- ◆ 在输入格式说明表中可使用连续多个斜杠编辑符，第1个以后的斜杠编辑符其作用是跳过一行。

2) 斜杠编辑符用于数据输出

在数据输出时，可使用斜杠编辑符将输出数据按多行输出。

```
      READ 10,i,a,j,b  
      PRINT 20,i,a,j,b  
10    FORMAT(I5,3x,f5.1/I5,3x,f5.1)  
20    FORMAT(1x,I5,3x,f5.1/1x,I5,3x,f5.1)  
      END
```

输入： 123456781234
 123456781234

输出： 12345 123.4
 12345 123.4

分析下列语句的运行结果

Write(*,"(13//13)")10,10

10

10

5.10 不可重复编辑符



◆ X
◆ /
◆ \
◆ T
◆ S
◆ :
◆ P
◆ BN
◆ Q



- \ 编辑符用于输出数据时不结束本行操作,下一输出语句接本行末输出。
- 其作用是用多个输出语句完成一行输出操作。
- 示例:

```
i=125;j=234  
PRINT 10,i  
PRINT 20,j  
10 FORMAT('i=',I3,\)  
20 FORMAT('j=',I3)  
END
```

输出: i=125j=234。





4. T、TL、TR编辑符

T、TL、TR编辑符可设置下一个数据的输入输出位置。

Tc 使下一个输入输出数据从输入输出行的第c个字符开始。

TLc 使下一个输入输出数据从当前位置向左移动c个字符后开始。

TRc 使下一个输入输出数据从当前位置向右移动c个字符后开始。

5. P、SS、S编辑符

SP、SS、S编辑符可控制“+”是否输出。SP、SS、S编辑符前后没有任何参数。

SP编辑符可控制正数在域宽内输出正号“+”，加号占一个字符位置。

SS编辑符可控制正数在域宽内不输出正号“+”。

S编辑符功能同SS编辑符，S编辑符为缺省编辑符。

6. :编辑符

冒号“:”编辑符可在没有要输出的数据项时结束数据输出。

纵向走纸控制

- ❖ 输出语句的格式 说明表的第**1**个编辑符总是**1X**编辑符,用于输出**1**个空格。**1X**编辑符除了输出空格外,还有一个特殊的功能,即“纵向走纸控制”。
- ❖ 通过显示器或打印机输出数据时,经常需要对行与行之间间隔进行控制,如一行接一行输出、隔行输出、换页输出等。
- ❖ 标准**FORTRAN**语言规定:将输出行中的第一个字符作为纵向走纸控制符,这个字符只起控制符作用,不再被输出。

纵向走纸控制符

控制字符	功能
空格	换行，将光标或打印头移到下一行开头
0	光标或打印机头前进二行
1	换页，将光标或打印头移到下一页开头
+	不换行，光标或打印头回到本行开头
其它字符	与空格功能相同

5.13 有格式输入输出应用举例



◆ 问题 ◆ 说明

输入3个同学4门课(数学、物理、化学、英语)成绩,统计每个同学总成绩和平均成绩,计算每门课平均成绩,输出每个同学每门课成绩、总成绩和平均成绩,以及数学、物理、化学、英语的平均成绩。

要求按以下格式输入成绩数据:

	数学	物理	化学	英语
输入第1位同学四门课成绩:	78.5	85.0	58.0	98.0
输入第2位同学四门课成绩:	75.0	88.5	99.0	45.5
输入第3位同学四门课成绩:	84.5	69.5	77.0	82.0

要求按以下格式输出有关数据:

	数学	物理	化学	英语	总成绩	平均成绩
第1位同学成绩:	78.5	85.0	58.0	98.0	319.5	79.875
第2位同学成绩:	75.0	88.5	99.0	45.5	308.0	77.000
第3位同学成绩:	84.5	69.5	77.0	82.0	228.5	57.125
数学平均成绩:	79.333分					
物理平均成绩:	81.000分					
化学平均成绩:	78.000分					
英语平均成绩:	75.167分					



5.13 有格式输入输出应用举例



◆ 问题

◆ 说明

→ 使用22个实型变量保存有关数据。设：

s11,s12,s13,s14：分别存放第1位同学4门课成绩。

total11,av11：分别存放第1位同学总成绩和平均成绩。

s21,s22,s23,s24：分别存放第2位同学4门课成绩。

total21,av21：分别存放第2位同学总成绩和平均成绩。

s31,s32,s33,s34：分别存放第3位同学4门课成绩。

total31,av31：分别存放第3位同学总成绩和平均成绩。

Av1,av2,av3,av4：分别存放数学、物理、化学和英语平均成绩。

(程序)



有格式输入输出应用程序

```
total11=s11+s12+s13+s14;av11=total11/4
total21=s21+s22+s23+s24;av21=total21/4
total31=s31+s32+s33+s34;av31=total31/4
av1=(s11+s21+s31)/3;av2=(s12+s22+s32)/3
av3=(s13+s23+s33)/3;av4=(s14+s24+s34)/3
WRITE(*,"(16X,'数学 物理 化学 英语 总成绩 平均成绩')")
WRITE(*,200) 1,s11,s12,s13,s14,total11,av11
WRITE(*,200) 2,s21,s22,s23,s24,total21,av21
WRITE(*,200) 3,s31,s32,s33,s34,total31,av31
200 FORMAT(1X,'第',I1,'位同学成绩: ',4(F4.1,2X),F6.1,2X,F7.3)
WRITE(*,300) '数学',av1,'物理',av2,'化学',av3,'英语',av4
300 FORMAT(1X,A4,'平均成绩:',F7.3,'分')
END
```

5.14 习题五



1. 何谓格式说明表？如何在有格式输入输出数据时指定格式说明表？
2. **FORMAT**语句的功能是什么？**FORMAT**语句可放在程序什么位置？
3. 下面是一些非法输入输出语句和格式说明语句。说明其非法原因。

READ * i,j,k

READ 10 i,j,k

READ *, 'i=', i, 'j=', j

READ(*, 10) i,j, 125

READ(*, "(i=', I3, 'j=', I3)") i,j

READ("(I3, I3)", *) i,j

10 FORMAT(1X, I3, :, I3, :, I3)

PRINT * i,j,k

PRINT 10 i,j,k

WRITE *, 'i=', i, 'j=', j

WRITE(20, *) i,j, 125

PRINT(*, "(i=', I3, 'j=', I3)") i,j

WRITE("(I3, I3)", *) i,j

20 FORMAT(1X, I3, BN, I3, BZ, I3)

4. **FORTRAN 90**提供了多少格式编辑符？格式编辑符划分为几类？
5. 判断下列格式编辑符。哪些是可重复编辑符？哪些是不可重复编辑符？

I、X、E、G、T、P、B、A、BN、Q、SP、EN、D、/、\$



5.14 习题五



6. 判断下列编辑符。哪些只用于输入?哪些只用于输出?哪些可用于输入输出?

I、X、E、G、T、P、B、A、BN、Q、SP、EN、D、/、\$

7. 给定下面READ和FORMAT语句:

READ 100,i,j,k,m,n

100 FORMAT(I2,I3,2X,I2,1X,I4.1,I3)

执行READ语句,输入以下5组数据:

12345678901234567890✓

1□322□43□□565□768□78✓

□12,3□43□7□□,2□3455✓

□□1232,45A18□93□215✓

1□23,324,445,555,777✓

READ语句执行后,每组i,j,k,m,n的值分别是多少?

5.14 习题五



8. 给定下面READ和FORMAT语句:

READ 100,a,b,c

100 FORMAT(F5.2,F5.1,E8.3)

执行READ语句,输入以下5组数据:

12345678901234567890✓

123,456789012,345890✓

1□322□43□□565□E+□28✓

123E2343□7□□2□3455✓

1.23245.E3.1893□215✓

READ语句执行后,每组a,b,c的值分别是多少?

输出

输出







5.14 习题五



14. 已知： $i=25, j=-125, k=-4725, m=-31500, a=48.57, x=-1.5E+11, y=158.895E-11$ 。

给定输出语句：WRITE(*,100)i,a,j,b,k,m,x,y

对下面不同的FORMAT语句,写出输出结果:

- 100 FORMAT(1X,I3,1X,F5.2,1X,I4,1X,F9.2,I5,I6,E12.3,E12.3E3)
- 100 FORMAT(3X,I4,1X,F9.2,1X,I4,1X,F9.2,2I5,2E12.3)
- 100 FORMAT('□i=',I3,'a=',F5.2,'j=',I4,'b=',F9.2,'k=',I5,'m=',I6,& 'x=',E12.3,'y=',E12.3E3)
- 100 FORMAT(2(2X,I4,2X,F9.2),2I5,2E12.3)
- 100 FORMAT(1X,I3,1X,F5.2/1X,I4,1X,F9.2/I5,I6,E12.3,E12.3E3)

15. 已知： $i=35, j=-425, k=4357$ 。

给定输出语句：WRITE(*,100)i,j,k

对下面不同的FORMAT语句,写出输出结果:

- FORMAT(1X,I4)
- 100 FORMAT(1X,2I4,2I7.5)
- 100 FORMAT(1X.2(I5,2X)/)

5.14 习题五



16. 已知: $i=315, j=512, a=43.57, b=-15.9, c=123.456, d=-5231.573$ 。

给定输出语句: `WRITE(*,100)i,j,a,b,c,d`

对下面不同的FORMAT语句,写出输出结果:

● `100 FORMAT(1X,I4,2X,I5,2X,2(F7.2,2X),F10.1)`

● `100 FORMAT(1X,2(I4,2X),2(F7.2,2X),'OLD')`

17. 已知: $i=78, j=158, x=85.5, y=542.953$ 。给定输入语句: `READ(*,100)i,x,j,y`

对下面不同的FORMAT语句,应如何输入数据:

■ `100 FORMAT(I4,F7.2,I5,F8.2)`

■ `100 FORMAT(I4,2X,F7.2,2X,I5,2X,F8.2)`

■ `100 FORMAT(1X,2(I4,2X,F8.2))`

■ `100 FORMAT(I4,F7.2/I5,F8.2)`

■ `100 FORMAT(I4/F7.2/I5/F8.2)`

■ `100 FORMAT(//I4,F7.2//I5,F8.2//)`

■ `100 FORMAT(I4,F7.2)`

■ `100 FORMAT(I4,F7.2,I5,F8.2,I6,F9.2)`

5.14 习题五



18. 给定下面程序：

```
READ 10,i,j,k,m,n
10  FORMAT(1X,I3,T5,I3,I4,TL8,I2,TR5,I3)
PRINT *,i,j,k,m,n
END
```

程序执行时,输入数据: 1234567890987654321✓

程序执行后,变量i,j,k,m,n的值是什么?

19. 编写程序输出以下字母图案:

```
      *                *****
    * *                *          *
  *   *                *          *
*****                *****
*       *            *          *
*       *            *          *
*       *            *****
```



5.14 习题五



20. 编写程序输出以下格式的九九乘法表：

1 2 3 4 5 6 7 8 9

1	1								
2	2	4							
3	3	6	9						
4	4	8	12	16					
5	5	10	15	20	25				
6	6	12	18	24	30	36			
7	7	14	21	28	35	42	49		
8	8	16	24	32	40	48	56	64	
9	9	18	27	36	45	54	63	72	81

PROGRAM PRINTA

```
write(*,"(7x,'*')")  
write(*,"(6x,'*',1x,'*')")  
write(*,"(5x,'*',3x,'*')")  
write(*,"(4x,'*****',x)")  
write(*,"(3x,'*',7x,'*')")  
write(*,"(2x,'*',9x,'*')")  
write(*,"(1x'*',11x,'*')")  
end
```

PROGRAM PRINTE

```
write(*,"(1 x,a)") repeat('*',12)
print *
write(*,"(1 x,a,1 1 x,a) '*','*")
print *
write(*,"(1 x,a,1 1 x,a) '*','*")
print *
write(*,"(1 x,a)") repeat('*',12)
print *
write(*,"(1 x,a,1 1 x,a) '*','*")
print *
write(*,"(1 x,a,1 1 x,a) '*','*")
print *
write(*,"(1 x,a)") repeat('*',12)
end
```



Th e

e n d

文件名格式：班级 学号 姓名 简略实验名称
邮件标题同文件名

Any questions please 发送至
xingzhengwu@163.com