

National Computer Rank Examination



全国计算机 等级考试

二级考试参考书 ——QBASIC 语言程序设计

教育部考试中心



高等教育出版社
Higher Education Press

全国计算机等级考试

二级考试参考书

——QBASIC 语言程序设计

教育部考试中心

高等教育出版社

H3101/01

图书在版编目(CIP)数据

全国计算机等级考试二级考试参考书——QBASIC
语言程序设计/教育部考试中心编. —北京: 高等教育
出版社, 2003. 3

ISBN 7-04-012679-6

I. 全... II. 教... III. ①电子计算机-水平考试
-自学参考资料②BASIC 语言-程序设计-水平考试-
自学参考资料 IV.TP3

中国版本图书馆 CIP 数据核字 (2003) 第 014135 号

责任编辑 肖子东 封面设计 王凌波 版式设计 马静如
责任校对 刘 莉 责任印制 韩 刚

出版发行	高等教育出版社	购书热线	010 - 64054588
社 址	北京市东城区沙滩后街 55 号	免费咨询	800 - 810 - 0598
邮政编码	100009	网 址	http://www.hep.edu.cn
传 真	010 - 64014048		http://www.hep.com.cn

经 销 新华书店北京发行所
印 刷 高等教育出版社印刷厂

开 本	850 × 1168 1/16	版 次	2003 年 3 月第 1 版
印 张	14.25	印 次	2003 年 3 月第 1 次印刷
字 数	350 000	定 价	23.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

前 言

全国计算机等级考试自 1994 年开考以来,适应了市场的需要,得到了社会的广泛认可,在推广普及计算机应用知识和技术,以及为用人单位录用和考核工作人员提供评价标准等方面发挥了重要作用。考试不是目的,而以考促学,为国家构建终身教育体系尽一份力量,才是全国计算机等级考试的最终目标。显然,全国计算机等级考试也是一种非学历的职业教育和继续教育形式。为了给广大考生提供更多的学习帮助和支持,在原有全国计算机等级考试教程的基础上,教育部考试中心组织编写了这套全国计算机等级考试考试参考书系列丛书。

本书是与教育部考试中心组织编写的《全国计算机等级考试二级教程——QBASIC 语言程序设计》相配套的学习参考书,各章的内容与教程相对应。本书每章包括四个部分:学习目标与要求、内容要点、例题分析与解答、自测题。各章在概括主要内容要点的基础上,对大量的例题做了分析和解答,同时编制了大量的自测题并给出了参考答案供考生练习和参考。

由于编写时间仓促,内容涉及面较广,疏漏之处在所难免,望读者提出宝贵意见,以便修订时改正。

编 者
2003 年 1 月

郑 重 声 明

高等教育出版社依法对本书享有专有出版权。任何未经许可的复制、销售行为均违反《中华人民共和国著作权法》。行为人将承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。社会各界人士如发现上述侵权行为，希望及时举报，本社将奖励举报有功人员。

现公布举报电话及通讯地址：

电 话：（010）84043279 13801081108

传 真：（010）64033424

E-mail: dd@hep.com.cn

地 址：北京市东城区沙滩后街 55 号

邮 编：100009

目 录

第一章	QBASIC 的基本概念	1
1.1	学习目标与要求	1
1.2	内容要点	1
1.3	例题分析与解答	9
1.4	自测题	12
1.5	自测题参考答案	13
第二章	顺序程序设计	14
2.1	学习目标与要求	14
2.2	内容要点	14
2.3	例题分析与解答	29
2.4	自测题	33
2.5	自测题参考答案	35
第三章	选择结构程序设计	37
3.1	学习目标与要求	37
3.2	内容要点	37
3.3	例题分析与解答	59
3.4	自测题	64
3.5	自测题参考答案	68
第四章	循环结构	70
4.1	学习目标与要求	70
4.2	内容要点	70
4.3	例题分析与解答	73
4.4	自测题	81
4.5	自测题参考答案	88
第五章	函数与子函数	89
5.1	学习目标与要求	89
5.2	内容要点	89
5.3	例题分析与解答	94
5.4	自测题	103
5.5	自测题参考答案	110
第六章	数组	111

6.1	学习目标与要求	111
6.2	内容要点	111
6.3	例题分析与解答	114
6.4	自测题	119
6.5	自测题参考答案	126
第七章	字符处理	127
7.1	学习目标与要求	127
7.2	内容要点	127
7.3	例题分析与解答	132
7.4	自测题	140
7.5	自测题参考答案	143
第八章	屏幕控制和作图	144
8.1	学习目标与要求	144
8.2	内容要点	144
8.3	例题分析与解答	146
8.4	自测题	150
8.5	自测题参考答案	155
第九章	文件	156
9.1	学习目标与要求	156
9.2	内容要点	156
9.3	例题分析与解答	159
9.4	自测题	166
9.5	自测题参考答案	170
第十章	上机考试	171
10.1	学习目标与要求	171
10.2	内容要点	171
10.3	例题分析与解答	172
10.4	自测题	187
10.5	自测题参考答案	200
附录 1	2002 年下半年全国计算机等级考试二级笔试试题——基础知识和 QBASIC 语言程序设计	203
附录 2	2002 年下半年全国计算机等级考试二级笔试试题参考答案——基础知识和 QBASIC 语言程序设计	218

第一章 QBASIC 的基本概念

1.1 学习目标与要求

- (1) 掌握 QBASIC 提供的数据类型;
- (2) 掌握 QBASIC 中常量和变量及其类型与命名规则;
- (3) 掌握 QBASIC 中运算符及其运算规则;
- (4) 熟悉 QBASIC 中表达式的概念, 并熟练地运用表达式求值。

1.2 内容要点

1.2.1 QBASIC 语言概述及程序结构

一、QBASIC 概述

QBASIC 是由 1964 年诞生的 BASIC 语言发展而来的, 它是 MicroSoft 公司开发的一种模块化和结构化的 BASIC 语言版本, 它不但继承了第二代 BASIC 语言版本的易学、易用等特点, 而且扩充了其功能, 主要表现在以下的几个方面:

- (1) 扩充了变量和常量的类型;
- (2) 提供了新的选择结构;
- (3) 改进了循环结构;
- (4) 提供了作为单独模块的子程序和函数;
- (5) 不再需要行号;
- (6) 具有良好的编辑环境;
- (7) 提供联机帮助;
- (8) 提供“分步执行”、“跟踪”等调试工具;
- (9) 采用先进的解释执行方式。

二、QBASIC 源程序结构

一个 QBASIC 程序由若干条语句或多个模块组成, 遵循如下规则:

- (1) 在一个程序中多个语句可以放置在一行中, 各语句之间以“:”分隔, 一行语句的长度不得超过 255 个字符。

(2) 语句行可以有行号也可以没有行号, 行号不起作用, 程序并不是按语句行号的大小先后执行, 而是按语句排列的顺序自上而下执行(行号为顺序执行的纯数字, 即由 1~40 个数字组成, 行号后面不加“:”)。

(3) QBASIC 程序还可以用“行标号”。行标号由 1~40 个字母或数字再加上一个“:”组成。如“10x:END”, “10x:”就是一个合法的标号(行号和标号必须写在每行语句的最左端)。

(4) 语句格式

语句用来“告诉”计算机完成什么操作, 一条语句一般包括如下所示的两部分:

语句定义符	语句体
-------	-----

① 语句定义符

例如: LET、PRINT、END (用来指定语句的功能)。

② 语句体

它跟在语句定义符后面, 用来告诉计算机需要执行的具体内容, 例如: $85*a$ 、 $70*b$ 等。QBASIC 中大多数语句都包含语句定义符和语句体这两个部分, 但也有少数语句不包含语句体。一个程序中可包含执行语句和非执行语句。如执行语句有: PRINT、IF、INPUT、END 等; 非执行语句有: REM、DATA 等。

(5) 每个程序通常以 END 结束 (执行程序时遇到 END 停止)。

一个 QBASIC 程序可以包含若干个子程序或函数, 如下:

① 主程序和每一个子程序均为一个独立的程序模块, 需分别进行设计。

② 程序执行时, 从主程序的第一行开始顺序执行, 遇到子程序时就调用子程序, 顺序执行该子程序中的语句, 执行完子程序后, 返回本次调用主程序处, 再接着执行主程序中后继的语句, 直到遇到 END 语句为止 (关于怎样进入编辑子程序以及有关子程序的细节, 将在第五章“函数与子函数”中详细介绍)。

注意:

每一个模块必须有一个结束语句, 主程序模块以“END”结束, 子程序模块以“SUB <子程序名称>”开始, 以“END SUB”结束, 函数以“FUNCTION <函数名称>”开始, “END FUNCTION”结束。

1.2.2 QBASIC 常量

程序处理的对象为数据, 程序的主要功能为指挥计算机自动地完成数据运算的全过程, 参加运算的数据称为运算对象, 也称“运算量”。QBASIC 中的运算量有: 常量、变量、数组和函数(标准函数、自定义函数等)。QBASIC 程序处理的数据分为数值型数据和字符型数据两大类。

常量是指在程序运行的整个过程中, 不能改变的数据。QBASIC 中的常量分为数值常量和字符常量两类。

一、数值常量

可以将数值常量理解为数学中的常数, 数值常量分为整型常量和实型常量两类。

1. 整型常量

所有的整数都是整型常量。QBASIC 中能够表示的有符号整数的范围是：-32 768~32 767(占用 2 个字节 16 位)。并且在 QBASIC 中整型常量只能由数字及其正负号构成，不包括其他的符号。例如：123、-10 000、+0、-0、-12345 都是合法的，而+45 000(太大)、-45 000(太小)、25 000(含“,”)都是非法的。

2. 实型常量

实型常量就是数学中的小数。它由数字、小数点、符号和指数构成。在 QBASIC 中有两种表示方法，即日常计数法和科学计数法。

(1) 日常计数法

即数学上的十进制小数形式，它由符号、数字和小数点构成。

例如：1.6、-3.02、0.0、-12、+3.150 等都是合法的 QBASIC 实数形式，这种形式的实数称为“定点实数”。因为这些实数中小数点的位置是固定的。例如，123.321 中小数点的位置在 3 之后，显然它不能改变为 1233.21，因为二者不是同一个常数。在 QBASIC 中，为了表示不同精度的定点数，又把定点数分为单精度型、双精度型两个数据类型。

(2) 科学计数法

即用指数形式表示的实数。例如，123.456 可以表示为 123.456×10^0 、 12.3456×10^1 、 1.23456×10^2 、 0.123456×10^3 等形式。在 QBASIC 中将用这种形式表示的数叫做浮点数，浮点数用有符号小数和指数的形式来表示，其中指数的表示形式如表 1.1 所示：

表 1.1 指数表示形式

指 数	
底数	幂
E(e)或 D(d)	不超过三位的有符号整数

注：D(d)专门用于表示双精度型浮点数。

123.456 可以表示为：+1.23456E+2、+123.456E0 等，如小数点位置改变，幂也要相应改变，如果小数部分或指数的幂为正数，就可以省略“+”。根据精度的不同浮点数又分为单精度型和双精度型两种。单精度型占用 4 个字节的存储空间，最多提供 7 位有效数字；双精度型占用 8 个字节的存储空间，最多提供 16 位有效数字。

单精度型数值常量用末尾带“!”的十进制定点数来表示，或用“E”(或“e”)来代表指数的底数。如 456.789 表示为 456.789! 或 4.56789E2。

双精度型数值常量用末尾带“#”的十进制定点数来表示，或用“D”(或“d”)来代表指数的底数。如 456.789 表示为双精度型是 456.789#或 4.56789D2。

说明：

- ① 纯小数可以省略小数点前面的 0，如-0.1 也可以表示为-.1。
- ② 有关常数类型的区分及表示范围请参照“教程”第 6~9 页 1.2.1 小节“数值常量”。

二、字符常量

字符常量是指用双引号括起来的一组合法的字符序列。一般情况下该序列需根据用途而确定。

例如, “ABCDE12345”、“A+B+C+D=”、“计算机等级考试”等都是字符常量。

三、符号常量

QBASIC 允许用户定义一个符号名来代表一个常量, 形式如下:

CONST 常量名 1=<常量 1>[, 常量名 2=<常量 2>, ..., 常量名 n=<常量 n>]

其中, 符号常量的命名规则与变量的命名规则相同。

符号常量名只是一个符号, 用它代表一个常量, 它不能被赋值, 下面语法是错误的。

```
CONST r=3.5
```

```
LET r=4.5
```

以上程序段显示出错信息, 因为 r 已定义为符号常量, 其值为 3.5, 不允许再改变, 采用符号常量可以免去重复书写一个常量, 大大减少出错概率; 使改变一个程序中的同一常量非常容易; 增加了程序的易读性, 使程序更易于维护, 并可以保证重要的参数不被改写。

1.2.3 QBASIC 变量

所谓变量是指在程序运行过程中, 其值可以变化的量。例如:

源程序:

```
LET a=3
```

```
PRINT a
```

```
LET a=5
```

```
PRINT a
```

```
LET a=8
```

```
PRINT a
```

运行结果:

```
3
```

```
5
```

```
8
```

在程序的运行中 a 先后有 3 个值, 也就是说 a 的值是可以变化的。

一、变量命名规则与变量值

1. 变量命名规则

每一个变量都应该有一个名字, 即变量名, 在 QBASIC 中对变量命名有如下的规定:

(1) 变量名以字母打头, 后跟字母、数字和小数点, 不能含空格并且不能含有“-”、“_”等符号, 1~40 个字符有效, 字母大小写均可。

(2) 表示变量类型的符号(如%、!、#、&、\$)应该放在变量名的最后。

(3) 不允许将系统的保留字(如 LET、RESTORE 等)作为变量名。

在定义一个变量时应考虑“见名思义”, 如可以用 average 来代表平均值。

2. 变量名与变量值

可用一个实例来理解变量名与变量值：假设一家旅馆有若干个单间，如图 1.1 所示，通过房间号可以找到每一个房间，任何一个空房间随时都可能有旅客入住，房号是不会改变的，而房间里面的旅客是时常改变的。假设没有新的旅客入住原来的旅客就不必搬走，如果在某一时刻，旅客张三入住 101 号房间，这时我们一见到 101 这一房号就会联想到张三。这里的房间号就相对于 QBASIC 中的变量名，而房间里面住的旅客相当于变量的内容即变量值。

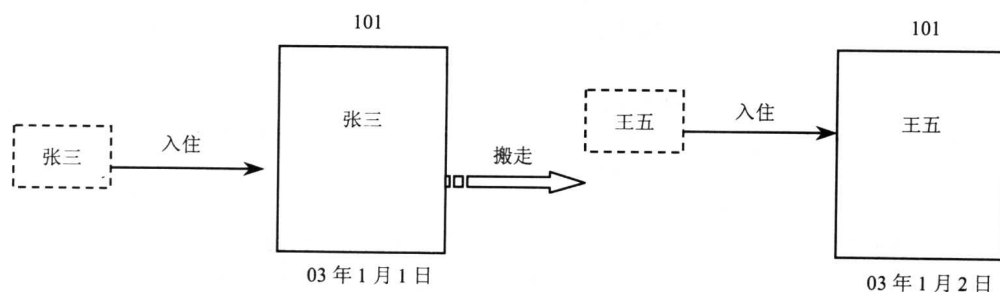


图 1.1 变量名与变量值实例

二、变量的类型

1. 每一个变量都是用来存储程序运行时的数据的，由于数据分为不同的类型，因此存放数据的变量也具有不同的数据类型。系统对不同类型的变量分配的存储空间也不一样，如表 1.2 所示。

表 1.2 变量类型定义

变 量 类 型	定 义	所占存储空间
整型变量	存放整数的变量	2 字节
单精度实型变量	存放单精度实数的变量	4 字节
双精度变量	存放双精度实数的变量	8 字节
长整型变量	存放长整型实数的变量	4 字节
字符串变量	存放字符串常量的变量	每个字符占 1 字节

2. 指定变量类型的方法

QBASIC 提供了三种方法来指定变量类型：

(1) 在变量名的末尾加上类型定义符

QBASIC 规定用 %、! 和 # 分别作为整型、单精度和双精度的类型符号。如写为 a%，则 a% 被指定为整型变量，b! 则表示 b! 被指定为单精度型变量，c# 则表示 c# 被指定为双精度型变量，d& 则表示 d& 被指定为长整型变量，e\$ 则表示 e\$ 被指定为字符型变量。

(2) 用 DIM 语句来定义变量并指定类型

格式：DIM <变量名> [AS <数据类型>]

例如，语句 DIM a AS INTEGER (定义 a 为整型变量，与 a% 相同)

QBASIC 的变量类型如表 1.3 所示。

表 1.3 变量类型

变 量 类 型	类型定义符	类 型 名	占 内 存	举 例
整型	%	INTEGER	2	a% 或 DIM a AS INTEGER
长整型	&	LONG	4	b& 或 DIM b AS LONG
单精度型	!(可有可无)	SINGLE	4	c! 或 DIM c AS SINGLE
双精度型	#	DOUBLE	8	d# 或 DIM d AS DOUBLE

例如:

DIM a AS INTEGER

'定义 a 为整型变量

DIM b AS SINGLE

'定义 b 为单精度型变量

DIM c AS LONG

'定义 c 为长整型变量

a=56:b=7.77:c=333333:PRINT a,b,c

end

运行结果:

56 7.77 333333

(3) 用 DEFtype 语句, 其中 type 表示类型

type 均为英文缩写, 其含义如表 1.4 所示。

表 1.4 type 的类型

数 据 类 型	英 文 缩 写
整型	INT
单精度型	SNG
长整型	LNG
双精度型	DBL
字符型	STR

例如, DEFINT a (代表凡以字母 a 打头的变量均为整型)

又如: DEFDBL e-g(e、f、g 开头的变量均为双精度型)

说明: 定义变量类型可以用上述三种方法的任一种, 但不能同时用这几种方法。

例如, DIM d# AS DOUBLE 便是错误的, 应改为 DIM d AS DOUBLE。

1.2.4 标准函数

QBASIC 语言提供了丰富的常用函数并把它们编成了子程序。作为系统函数提供给用户使用, 用户在使用时只需要写出它的名字(即函数名)和形式参数就可以直接引用。具体形式如下:

函数=函数名([形式参数])

说明:

- ① 其中形式参数为可选项, 它可以为常量和变量。有些函数不需要形式参数。
- ② 调用函数后而得到的值称为函数的返回值。

QBASIC 中的系统函数见表 1.5。

表 1.5 QBASIC 中系统函数

函 数 种 类	数 量
字符串函数	16 个
有关文件函数	11 个
有关输出函数	16 个包括图形、声音、时间
其他函数	7 个
数学函数	18 个

详细说明见教程附录。

常用的数学函数请参阅“教程”第 14 页的表 1.4。

说明:

- ① 在使用函数时一定要注意函数中的参数。如数学函数中的三角函数的参数要求为弧度, 而不是角度。

例如, 在程序中调用函数求 $\sin 45^\circ$ 的值, 应采用如下的语句:

`SIN(45*3.14/180)`

- ② 注意函数所代表的含义。

例如, $\text{LOG}(X)$ 不代表数学上的 $\log_{10} x$ 而代表 $\ln x$, 因此若要计算 $\lg 13$ 的值, 应采用换底公式 $\lg x = \ln x / \ln 10$, 可表示为如下形式: $\text{LOG}(13)/\text{LOG}(10)$

- ③ 在进行混合运算时一定要注意函数的返回值。

例如, `a%=sin(0.5)`

`sin(0.5)` 的返回值为实型, 不应该存储在整型变量中。

1.2.5 运算符和表达式

表达式是由常量、变量和函数及括号通过特定的运算符连接起来的式子。

在 QBASIC 中有三类运算符, 它们是用来进行算术运算的算术运算符; 用来比较两个运算量大小的关系运算符; 用来进行逻辑运算的逻辑运算符。与此相应 QBASIC 的表达式也有三类, 即算术表达式、关系表达式、逻辑表达式。

一、算术运算符

QBASIC 中的算术运算符不仅包括大家熟知的加、减、乘、除(依次对应 QBASIC 中的 +、-、*、/)外, 还包括整除符号 \, 求余运算符 MOD 和乘方运算符 ^。

说明:

- ① 整除运算符 \

如果参加运算的两个量都是整数, 则取其相除后商的整数部分。例如, 表达式 $1 \setminus 3$ 的运算结果为 0, 如果参加运算的量为实数, 则先将它(它们)按四舍五入的原则变为整数, 然后再进行相除取整。例如, 表达式 $123.456 \setminus 5.4$ 的运算结果为 24。系统在运算前首先将 123.456 按四舍五入原则变成 123 并且把除数 5.4 变成 5, 然后用 123 除以 5 得出商, 最后取商的整数部分, 则结果为 24。

② 求余运算符 MOD

如果参加运算的两个量都是整数则系统将这两个量相除, 取运算后的余数。如表达式 $8 \text{ MOD } 3$ 的运算结果为 2; 与整除类似, 如果参加运算的量为实数, 则先将它(它们)按四舍五入的原则变为整数, 然后再相除取其余数。如表达式 $-123.456 \text{ MOD } 5.4$ 的运算结果为 3, 系统在运算前首先将 -123.456 按四舍五入原则变成 -123, 5.4 变成 5, 然后用 -123 除以 5, 取出余数。如果被除数与除数的符号不一致, 则余数的符号与被除数相同, 余数的绝对值为两个正数相除的余数。例如, 表达式 $15 \text{ MOD } -7$ 的运算结果为 1, 被除数是正数, 故余数也应为正数, 15 除以 7 取余数结果为 1。

③ 乘方运算符 ^

数学中的 x^n 在 QBASIC 中表示为 $X \wedge n$, 如 2^3 可以表示成表达式 $2 \wedge 3$, 运算结果为 8。运算规则与数学中一致。

二、算术表达式

在 QBASIC 中用算术运算符和括号将若干个数值型运算量(包括常量、变量、函数等)连接起来的式子称为 QBASIC 算术表达式。它的值也是一个数值型的数据。当多个运算符号出现在一个表达式中, 就需要规定各运算符的优先级, QBASIC 的运算规则如表 1.6 所示。

表 1.6 QBASIC 的运算规则

优 先 级	运 算 符	说 明
1	()	括号
2	+, -	正、负号(单目运算符)
3	函数	函数
4	^	乘方运算
5	*, /	乘、除运算
6	\	整除运算
7	MOD	求余运算
8	+, -	加、减运算

另外 QBASIC 还规定对于同级运算符按照从左到右的顺序进行运算。例如, 表达式

$$(3+5)*5 \text{ MOD } 2 + 2*3*\text{SQR}(4)^2$$

的运行结果为 24。

具体执行过程如下:

上 一 状 态	运 算 规 则	状 态
$(3+5)*5 \text{ MOD } 2 + 2*3*\text{SQR}(4)^2$	括号的优先级为 1, 先算括号里表达式的值	$8*5 \text{ MOD } 2 + 2*3*\text{SQR}(4)^2$

续表

上一状态	运算规则	状态
$8*5 \text{ MOD } 2 + 2*3*\text{SQR}(4)^2$	函数优先级为 3 计算函数值	$8*5 \text{ MOD } 2 + 2*3*2^2$
$8*5 \text{ MOD } 2 + 2*3*2^2$	乘方优先级为 4 计算乘方	$8*5 \text{ MOD } 2 + 2*3*4$
$8*5 \text{ MOD } 2 + 2*3*4$	乘号优先级为 5 计算乘法 按照从左到右的顺序计算	$40 \text{ MOD } 2 + 24$
$40 \text{ MOD } 2 + 24$	求余运算优先级为 7, 先求余	$0 + 24 = 24$

注意:

在书写表达式时要严格遵守运算规则。例如, 计算 8 除以 3 的余数的平方, 把表达式写成 $8 \text{ MOD } 3^2$ 就无法得到正确的结果, 因为乘方的优先级大于取余, 这样表示就成了计算 8 除以 $9(3^2$ 的值为 9) 的余数, 所以应加括号, 以保证先作求余运算, 故正确的表示形式应该是: $(8 \text{ MOD } 3)^2$ 。

三、不同数据类型的混合运算

在一个算术表达式中可能含有不同类型的运算量, 如: $123.456*5.4\text{D}10$ (单精度与双精度), $12345678/12$ (长整型与整型)。对于不同类型的混合运算 QBASIC 有如下规则:

(1) 同类型的数据进行算术运算, 结果值的类型不变。例如: $2*25*120=6000$ 。

如果两个整型运算量的乘积超过 $-32768 \sim 32767$ 的范围, 则显示出错“OVERFLOW”(溢出)。

(2) 整型数据与单(双)精度型数据进行运算, 其结果为单(双)精度型, 如: $315+86.3456789\#$ 结果为双精度型。

(3) 长整型与单精度型或双精度型运算, 结果为双精度型, 如: $1036372536\&*3.14!$ 结果为双精度型。

(4) 整型与长整型运算的结果为长整型。

(5) 单精度型与双精度型运算的结果为双精度型。

说明:

① 运算时, 先把数据类型级别低的数据转换成级别高的数据, 然后进行两个同类型数据的运算。例如: $12*26.5$, 先将 12 转换成 12.0, 然后 $12.0*26.5$ 运算为单精度的(或者说: 结果为级别较高的类型)。

② 级别高是指能够表示的数范围广, 占用的存储空间大。

③ 运算量包括变量、常量、数组和数组元素、函数等。

注意:

数值型数据与字符串数据不能进行混合运算。例如: `PRINT 123+"123"` 是非法的。

1.3 例题分析与解答

例 1 下列关于 QBASIC 的叙述中哪些是正确的?

① 符合 ANSI 标准

② 采用先进的解释执行方式并且可以采用编译的方式生成 EXE 文件

- ③ 可以不写行号
 ④ QBASIC 这一功能强大的编程工具需要占用较大存储空间
- A. ①②④ B. ②③
 C. ①②③ D. 全部

【分析】 本题考查的是关于 QBASIC 特点的知识。

1964 年美国的两位科学家创造了 BASIC 语言, 这种语言具有易学、易用的特点, 得到了广泛的推广。初期的 BASIC 由于当时计算机容量的限制, 只是一种小型语言, 在小型机上使用分时系统, 采用编译方式, 不仅能用于学习, 也能用于解决小型机问题。美国标准化协会(American National Standard Institute)1974 年制定了第一个 BASIC 标准 Minimal BASIC。BASIC 在不断地发展, 1987 年美国微软公司推出了 Quick BASIC, 它提供了一个开发程序的集成环境, 今天使用的 QBASIC 就是由其发展而来, QBASIC 是一种模块化和结构化的 BASIC 语言版本, 它与前一代 BASIC(如 GW BASIC)完全兼容, QBASIC 开发工具占用的磁盘空间较少, 一张软盘足以存储。与前一代 BASIC 相比 QBASIC 在很多方面扩充了功能, 其中包括不需要行号、用先进的解释执行方式并且可以采用编译的方式生成 EXE 文件等。

【答案】 C

例 2 在下列 4 个选项中可以作为 QBASIC 中的合法变量名的是

- A. TAN B. X.1
 C. 1X D. X -1

【分析】 本题考查的是关于 QBASIC 中变量命名规则的知识。

QBASIC 中的变量命名规则中要求变量名是唯一的; 不可使用保留字作为变量名; 不得以数字开头; 变量中不得出现空格、“-”及“_”等; 变量名中第 2 个字符及其后的各字符可以是字母、数字或小数点等; TAN 是 QBASIC 的保留字不能作为变量名(具体规则参考命名规则部分)。

【答案】 B

例 3 在 QBASIC 中, 与 a=396# 等价的表达式为

- A. a=3.96D2 B. a= 3.96E2
 C. a=396! D. a=396%

【分析】 本题考查的是关于数值常量表示方法的知识。

数值型常量分为整型常量和实型常量两类。实型常量分为单精度和双精度两种类型, 有两种表示方法, 日常计数法和科学计数法。一个常量的类型由后面的符号指定, 其中%用指定整型数据, 用&指定长整型数据, 如用日常计数法表示, 实型常量则用!来指定单精度, 用#来指定双精度; 如用科学计数法表示(表示成浮点数), 用数值部分和指数部分来表示, 指数部分用 E(e)/D(d)的形式来表示, 其中 D 为双精度型专用。

本题中常量后面的指定符号为#表示为双精度型, 将 396 按科学计数法表示为双精度型应为 3.96D2。

【答案】 A

例 4 下面的两段程序正确吗? 如正确写出程序的结果, 如不正确请说出为什么。

程序 1

CONST c = 20

程序 2

CONST a# = 20