

C 语言期末急救

·有其他编程语言/C 语言基础的同学可酌情食用
·无基础冲刺期末六十分大关的同学请认真阅读并仔细食用(可能不好吃...喵喵喵?)
·如果你还有两个小时就要考试, 请迅速翻阅本资料中以下划线或*星号*标记的重点

目录

第一部分 程序设计和 C 语言	2
计算机语言的发展阶段:	2
高级语言的发展阶段	2
第二部分 数据类型	3
常量分类:	3
变量	3
*数据类型:	3
第三部分 C 运算符	4
常用运算符类型	4
第四部分 C 语句	4
控制语句	4
函数调用语句	4
表达式语句	4
第五部分 基本输入输出	5
*printf 函数的一般格式	5
*scanf 函数的一般格式	5
*格式字符	5
第六部分 基本语句使用	6
*声明变量	6
if 语句	6
for 循环	6
while 循环	6
do while 循环	6
*while 和 do-while 循环的区别:	6
关于 continue 和 break 语句:	6
第七部分 逻辑运算符和逻辑表达式	7
*关于逻辑运算符	7
含义	7
优先次序	7

祝各位 rp++, 顺利通过考试!

第一部分 程序设计和 C 语言

计算机语言的发展阶段：

- 机器语言 machine language
 - 采用二进制代码(0 和 1)构成的机器指令
 - 与自然语言差别大，难以学习
- 汇编语言 assembler language
 - 采用助记符(例如 ADD、SUM)
 - 相比机器语言利于记忆
 - 不同型号的计算机的机器语言和汇编语言是互不通用的
 - 也称为低级语言
- 高级语言 high level language
 - 第一个计算机高级语言 FORTRAN
 - 需要使用编译器(compiler)将源程序(source program)转换为目标程序(object program)

高级语言的发展阶段

- 非结构化的语言
 - 风格随意，流程可以随意跳转
 - FORTRAN、早期 BASIC 等
- 结构化语言
 - 程序必须由基本结构构成(顺序结构、选择结构、循环结构)
 - 流程不允许随意跳转，由上而下顺序执行
 - C 语言、QBASIC 等(面向过程)
- 面向对象的语言
 - 支持面向对象程序设计方法
 - C++、C#、Java、Visual Basic 等

C 语言的发展

- 1972 年在 B 语言基础上设计出了 C 语言
- 两次标准修订：C89 和 C99(1989 年和 1999 年)

名词解释：

- 源程序，目标程序，可执行程序
- 程序编辑，程序编译
- 函数，主函数，被调用函数，库函数

第二部分 数据类型

在计算机高级语言中，数据有两种表现形式：

常量和变量

常量分类：

- 整型
- 实型
- 字符(普通字符、转义字符)
- 字符串
- 符号(#define)

变量

必须先定义、后使用

*数据类型：

基本类型：

整型

·基本整型(int)

·字符型(char)

浮点型

·单精度浮点型(float)

·双精度浮点型(double)

空类型(void)

派生类型：

·指针类型(*)

·数组类型([])

关于字符型数据

基本 ASCII 码(美国信息交换标准代码)包含了 127 个字符，使用 7 位二进制包含了大小写英文字母、数字、专门符号、空格符等

特别提醒：

大写字母'A'的 ASCII 码是十进制数 65

小写字母'a'的 ASCII 码是十进制数 97

第三部分 C 运算符

常用运算符类型

算数运算符: + - * / % ++ --

关系运算符: > < == != <= >=

逻辑运算符: ! && ||

条件运算符: ? :

第四部分 C 语句

控制语句

if() else 语句	条件语句
for() 语句	循环语句
while() 语句	循环语句
do while() 语句	循环语句
switch case 语句	多分枝选择语句
continue	结束本次循环
break	终止执行 switch 或循环语句
return	从函数返回

函数调用语句

函数名(参数);

例如:

```
printf("hello world");  
max(a,b);
```

表达式语句

表达式;

例如:

```
a=1;  
i=i+1;  
i++;
```

第五部分 基本输入输出

要使用 printf 和 scanf，需要包含头文件 stdio.h

即将 #include<stdio.h> 指令放在程序的开头

*printf 函数的一般格式

printf(格式控制,输出表列);

例如: printf("%d",a);

printf("%d %lf",a,b);

显而易见地, a 是一个整型变量, b 是一个双精度浮点型变量

当 a=1,b=2 时, 这两行代码的输出分别为:

1

1 2.000000

*scanf 函数的一般格式

scanf(格式控制,地址表列);

例如: scanf("%d",&a);

scanf("%d%lf",&a,&b);

由于 scanf 需要的参数是变量地址, 而不是变量名, 因此需要写成"&a"而不是"a", 这是 scanf 和 printf 最明显的区别。

*格式字符

整型 %d

字符 %c

浮点数 %f、%lf

字符串 %s

第六部分 基本语句使用

*声明变量

整型 int a;
字符 char b;
整型一维数组 int c[10];
整型二维数组 int d[10][10];

if 语句

```
if(条件 A){  
    满足条件 A 时运行的代码  
}else if(条件 B){  
    不满足 A 但满足 B 时运行的代码  
}else{  
    条件 AB 都不满足的时候运行的代码  
}
```

for 循环

```
for(初值;终止条件;步长){  
    循环体  
}
```

while 循环

```
while(条件){  
    循环体(满足条件时会一直循环运行的代码)  
}
```

do while 循环

```
do{  
    循环体  
}while(循环控制表达式);
```

*while 和 do-while 循环的区别:

while 循环是先判断条件，然后再进入循环
do-while 循环是先执行一次循环体，再进行条件判断

关于 continue 和 break 语句:

continue 是跳过本次循环，例如在 for 循环中跳过 i=2 时的这次循环，直接从 i=1 跳到 i=3。

break 是跳出整个循环语句，当 break 语句被运行时，立刻终止整个循环的运行。

说人话版: continue 只能跳过接下来这一次循环，break 要跳过接下来的整个循环

第七部分 逻辑运算符和逻辑表达式

*关于逻辑运算符

&&	逻辑与(AND)
	逻辑或(OR)
!	逻辑非(NOT)

含义

$a \& \& b$

如果 a 和 b 都为真，则结果为真，否则为假

$a \parallel b$

只要 a 和 b 中有一个为真，结果就为真；当二者都为假时结果才为假

$!a$

如果 a 为假，则!a 为真；如果 a 为真，则!a 为假

优先次序

!(非) -> &&(与) -> ||(或)