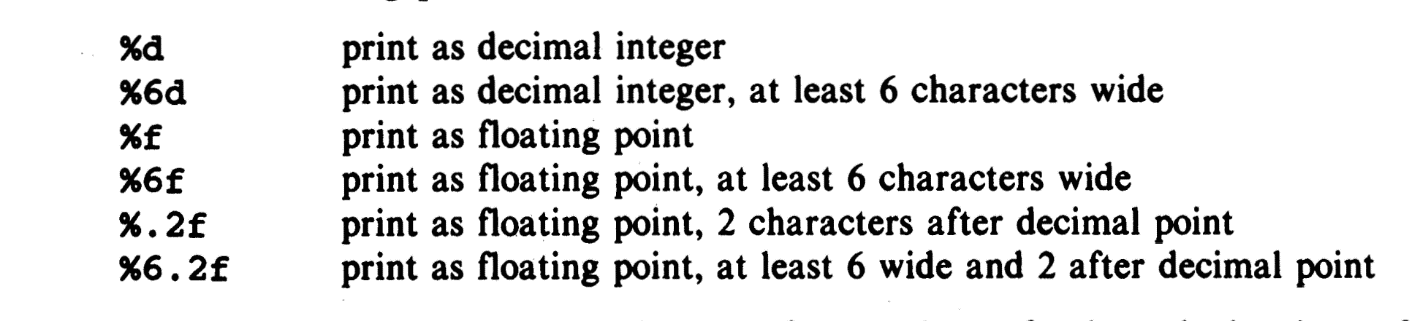
**2017/8/23自学**

**Printf的使用**

****

**Define的使用**

**在main前面define**

**#define OUT 0**

**不用加;**

**Terminal的快捷键**

**Ctrl+ C 结束进程**

**Ctrl +Z 模拟EOF**

**Ctrl+/ 或者Ctrl+Shift+/ 快速Comment一行或者一段**

**程序名称<input.txt**

**Vim的快捷键**

**gg=G auto format**

**Character**

**‘ ’ 空格**

**\n Newline Character is also a character**

**\b is backspace is also a character**

**\\ is backslash**

**‘\0’ null terminated string**

**在gdb时出现**

**Single stepping until exit from function main,**

**which has no line number information.**

**可能是gcc和gdb的版本新旧不兼容**

**或者试试gcc -g**

**Define Function步骤**

**int func(int a); //Function prototype**

**int main(){ //Call Function }**

**int func(int a){ //Function Definition }**

**Tricks**

**使用连等来initiate大量的variables**

**int a, b, c;**

**a = b = c = 0;**

**Function的运行不会直接overwrite外部varible的value**

**除非是pointer（Array 也是一种pointer）**

**比如**

**int func(int n){**

**n = 0;**

**return n;}**

**int n= 5;**

**结果是**

**func(n) 是0;**

**n =还是5;**

**如果需要input的Variable根据程序发生改变，input的需要是个pointer**

**所以如果argument中使用的是array（也是一种pointer），程序会直接改写Variable**

**"Definition" refers to the place where the variable is created or assigned storage;**

**"declaration" refers to places where the nature of the variable is stated but no storage is allocated**

**Variables + operators = expression**

**2017/8/23 LAB NOTE**

**-o Rename the executable gcc-o hello hello.c**

**-std=gnu99 Compiles against the GNU C99 standard, which we will use in this course.**

**-g Enables debug information, allowing the use of a debugger (gdb) on the executable**

**-Wall Enables all warnings**

**-Werror Treats all warnings as errors**

**Source file and Object File**

**An object file is just what you get when you compile one (or several) source file(s).**

**It can be either a fully completed executable or library, or intermediate files.**

**The object files typically contain native code, linker information, debugging symbols and so forth.**

**-c to compile to an object file**

**Source file >> Object file >> executable**

1. **The C compiler first compiles source files to object files, and checks that they are internally correct**
2. **Then gcc links the object files together**
3. **Gcc by default both compiles and links files for you**
4. **gcc doesn’t need to know about add.h directly, the #include directive in exp\_short.c is all that is required.**

**Switch**

**A switch statement allows a variable to be tested for equality against a list of values. Each value is called a case, and the variable being switched on is checked for each switch case.**

**switch(expression) {**

**/\***The **expression** used in a **switch** statement must have an integral or enumerated type, or be of a class type in which the class has a single conversion function to an integral or enumerated type.**\*/**

**case constant-expression : //value to compare to expression + colon**

**statement(s);**

**break; /\* 如果不要继续运行底下的case了，break掉\*/**

**case constant-expression :**

**statement(s);**

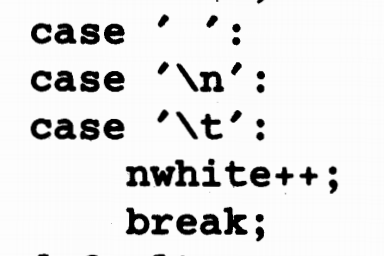
**break;**

**default : /\* 不适用上面的case的其他情况在这里，不需要break \*/**

**statement(s);**

**}**

**相同statement的例子**

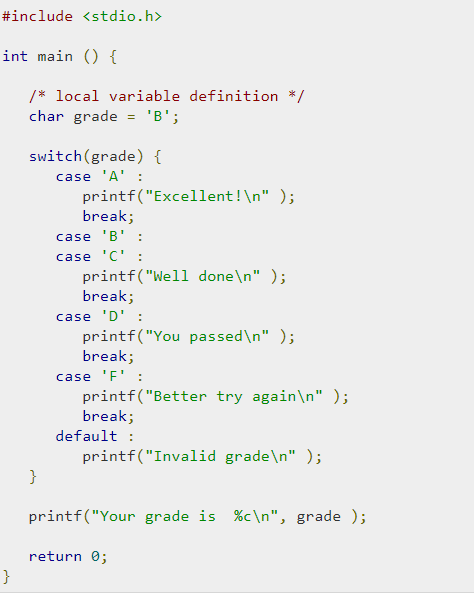


**How this works**

If a case matches the expression value, execution starts from that case

If no case matches, no execution will be going on

**例子**



**String的functions**

**Strcpy(char \*dest, char \*src)-Copy string src to dest**

**Int strlen(char \*s)**

**Char \*strcat(char \*dest, char \*src)**

**Strcat will not malloc for you, need to make the string large enough to have both dest and src and the null terminator.**

**Int strcmp(char \*a, char \*b)-Return positive if a is larger, negative if b is larger.**

**char \*strdup(const char \*s)-return a duplicate of the string s in memory allocated using malloc**

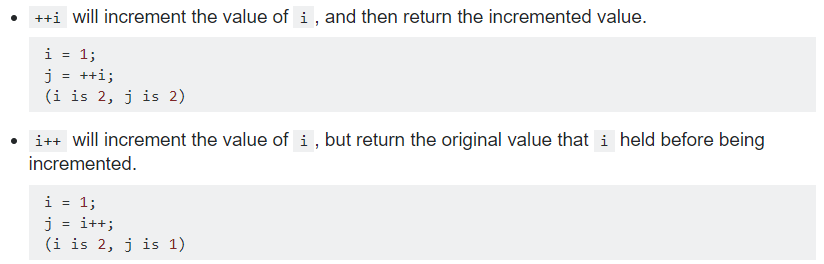
**return的string不能add up more elements bc the size is malloced**

**char \*strncpy(char \*dest, const char \*src, size\_t n)**

**Copy at most n bytes from string src to dest,**

**returning a pointer to the start of dest**

**i++和++i**



**除数和余数**

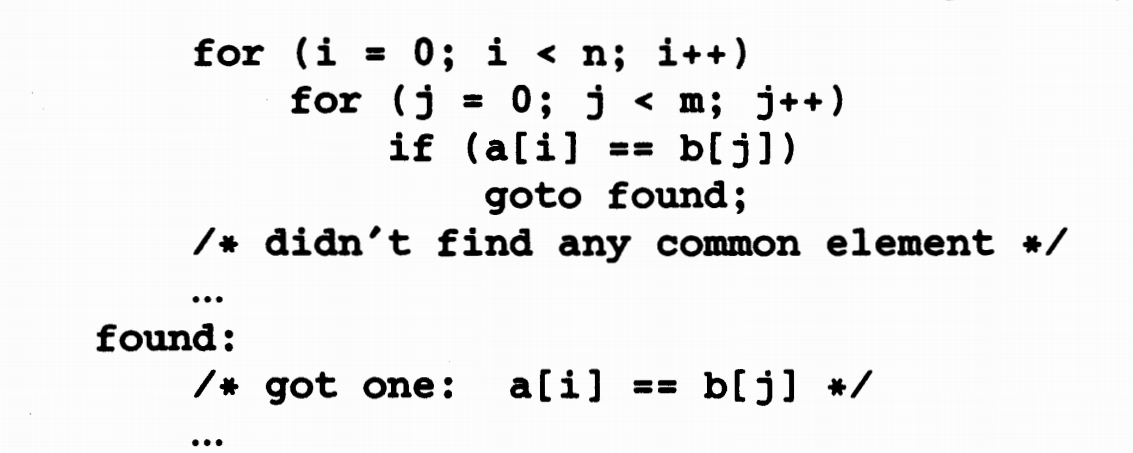
**10%3 = 1 余数**

**10/3 = 3 除数**

**Break,GoTo和Continue**

The break statement provides an early exit from **for, while, and do**, just as from **switch**.只能break innermost的loop

如果有很多层的loop，都需要跳出，可以



Continue;跳过循环后面的statement 直接进行下一层循环

怎么读取argv的文件input？Fopen,fget

char \* fileName = argv[1]; //保存filename

FILE \* fd = fopen(fileName, "r"); //打开filename并存到FILE类型的pointer里

如果 fd是NULL，printf没找到文件

Fgetc(fd) 和getchar一个功能

Print pointers 0x%lx

Point pointers as unsigned decimals %lu