

## Tutorial 1 – Basic C Programming and Control Flow

1. State the data type of each of the following:

a. '1'	char	g. 1870943465324L	long
b. 23	int	h. 1.234F	float
c. 0.0	float	i. -564	int
d. '\040'	non-printable char	j. 0177	int
e. 0x92	int	k. 0XfF4	long int
f. '\a'	non-printable char	l. 0xaaBB76L	long int

2. (a) What will the following program output? (refer to an ASCII table)  
 (b) What will happen if the format specifier of the second printf is changed to %d?  
 (c) What will be the result if **0x** in the third printf is removed?  
 (d) What if the first **0** in the fourth printf is deleted?

```
#include <stdio.h>

int main()
{
    printf("%c", 'A');
    printf("%c", 65);
    printf("%c", 0x41);
    printf("%c", 0101);
    return 0;
}
```

A  
A  
A  
~~5~~ A

A 65 A A

3. Assume x and y are integer variables. What will happen if one of the following statements is executed?

- (a) scanf("%d %d", &x, &y);  
 (b) scanf("%d %d", x, y);  
 (c) scanf("%d/%d", &x, &y);

x y  
error  
x y

4. The output of the following code is not zero. Why?

```
{
    .....
    double A = 373737.0;
    double B;

    B = A * A * A + 0.37/A - A * A * A - 0.37/A;
    printf(" The value of B is %f.\n", B);
}
```

double and float can be inexact

5. Given the following declarations and initial assignments:

int i, j, m, n;

float f, g;

i = j = 2;  
m = n = 5;  
f = 1.2;  
g = 3.4;

evaluate the following expressions independently, i.e. all variables start with the same set of initial values. Show any conversions which take place and the type of result.

- (a)  $m * j / j$   $5 \times 2 / 2 = 5$  int ✓  
 (b)  $m / j * j$   $5 / 2 \times 2 = 5$  int ✓  
 (c)  $(f + 10) * 20$   $(1.2 + 10) \times 20 = 224.0$  float ✓  
 (d)  $(i++) * n$   $3 \times 5 = 15$  int ✓  
 (e)  $i++ * n$   $2 \times 5 = 10$  int ✓  
 (f)  $-12L * (g - f)$   $-12 \times (3.4 - 1.2) = -26.4$  float ✓  
 (g)  $m = n = -j;$   $m = n = 1$  int ✓  
 (h)  $(int) g * 10$   $3 \times 10 = 30$  int ✓  
 (i)  $(int) (g * 10)$   $3.4 \times 10 = 34$  int ✓  
 (j)  $j = i + f$   $j = 2 + 1.2 = 3.2$  float ✓

6. Which of the following are acceptable case constant expressions? Assume the convention that upper case is used for defining a constant, e.g.

#define SVALUE 10

and other identifiers are variables.

- (a) case 76: ~~no~~ valid ✓  
 (b) case number\*2: No ✓  
 (c) case SVALUE\*2: yes ✓  
 (d) case 80.1: No ✓
7. In some computer games it is necessary to introduce a delay to slow the computer down. Assume that you are running the following program on a computer which uses 16 bits to represent an integer. How can the delay be (a) shortened, (b) made a thousand times longer, (c) made variable after compilation?

```
#include <stdio.h>
#define DLENGTH 32000

int main()
{
    int count;
    .....
    for (count = -DLENGTH; count <= DLENGTH; count++)
        ; /* this is a NULL statement which does nothing */
    .....
}
```

*Handwritten notes:*  
 while (count != DLENGTH) time+=1;  
 count++ in for loop  
 .11f printf()

8. Are the following code segments the same?

- (a) if (x != 0 && 2/x != 1) { .....}  
 (b) if (2/x != 1 && x != 0) { .....}

No

9. Write a section of C program to interchange the values of two integer variables. Is there a way of solving this problem without using a third variable?

```
#include <stdio.h>
int main(){
    int i, h;
    i = h+i;
    h = i - h;
    i = i - h;

    printf("i = %d, h = %d", i, h);
    return 0;
}
```