

Tutorial 1 – Basic C Programming and Control Flow

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

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

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;
}
```

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);`

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);
}
```

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 | (b) m / j * j |
| (c) (f + 10) * 20 | (d) (i++) * n |
| (e) i++ * n | (f) -12L * (g - f) |
| (g) m = n = --j; | (h) (int) g * 10 |
| (i) (int) (g * 10) | (j) j = i + f |

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: | (b) case number*2: |
| (c) case SVALUE*2: | (d) case 80.1: |
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 */
    .....
}
```

8. Are the following code segments the same?
- (a) **if (x != 0 && 2/x != 1) {}**
- (b) **if (2/x != 1 && x != 0) {}**
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?