**Lab Sheet 3**

1. Write two programs which execute a division by zero, one program which uses int variables and one using float variables. See what happens when the programs are run.
2. Type in, compile and run the following program.

**#include <stdio .h>**

**main ()**

**{**

**int x = 3, y = 5 ;**

**printf (“X: %d \n Y : %d\n” , x,y) ;**

**}**

**In the next several exercises you will introduce several syntax errors into this program, in order to examine the compiler error messages.**

3) Introduce an error by replacing the comma in the variable declaration with a semicolon. The compiler will now think you are trying to use the variable y without declaring it. Compile the new program.

4) Replace int with Int and recompile the program.

5) Delete the second double quote from printf()’s format string. Compile the new program.

6) Replace the variable declaration line with the following:

int x = 3, y = 5, y;

Compile the new program.

7) Remove the semicolon from the line containing the printf () statement and recompile.

8) Remove the final closing curly brace. Compile the new program.

9) Declare a character variable, assign it the value ‘X’ , and print out the variable using printf().

10) Declare a character variable, read a value into the variable using scanf () and print it out

using printf().

1. Print out the values of the character codes for the charaters ‘A’ ,‘Z’,’a’,’z’, ‘O’, ‘9’, and ,\n’. Use printf () with the %d format to print the values.
2. Write a program which uses the sizeof operator to compute the number of bytes used by each type ( char, short, int, long, float, double, long double) on your machine.

?gws\_rd=cr&ei=xNXPUu2mFMGHrAelmIHwAw#q=grep+command+in+linux+with+examples

13) Write a program which reads in a **float** from the user, tests if the value is positive, and if so, computes the square root of the value and prints out the result. Use the library function **sqrt()** to compute the square root and %f printf() format to print out the result. Put the following line in your program to include information about the **sqrt()** function

14) Write a program which reads in two **floats** and prints the maximum of the two using the ternary operator.

15) Write a program which reads in a digit (0-9) and prints out the decimal equivalent of that digit. Use getchar() to read in the digit and printf () to print out the result .Remember that get char() returns the character code for the digit, so subtracting the character code for ‘0’ from the digit you read in will give you the integer value you need. For example, if you read in the character ’5’ you will really get the character code for ‘5’ (which is ASCII 53). Subtracting ‘0’ (ASCII 48) will give the integer value 5.

16) Write a program to print the largest between two numbers with the use of ternary

operator ( **? :** )

17) Recall that a comparison results in a numeric value: 1 if the comparison is true and 0

if it is false. Write a program which prints out the values of the following comparisons.

• **5 >= 10**

• **( 5 < 10 ) && ( -37 > -40 )**

18) Write a program which prints out the value of an expression of the following form:

**0 < i < 10**

where i is an integer variable. The expression is legal C syntax but is not the correct way to test for a range of values. If the value of i were 15, would the expression be true? What is the reason?

19 Write a program to calculate the distance between two points. You may ask the user to input any two points.

20 . Write a program to find the largest of three numbers using ternary operator.

21 . Write a program to read a character in upper case and then print it in lower case.

22 Type the following programs and see how it works:

i) #include<stdio.h> //To understand circumflex operator ^

main()

{

char line[100];

printf("Any sentence");

scanf("%[^\n]",line); /\* enter any sentence \*/

printf("%s", line);

}

ii) #include<stdio.h>

main()

{

char line[100];

printf("Any sentence");

gets(line);

puts(line);

}

iii) /\* maximum field width specification \*/

#include<stdio.h>

main()

{

int a,b,c;

printf("enter any three numbers");

scanf("%3d%2d%3d", &a, &b, &c);

printf("%d\n%d\n%d\n" ,a,b,c);

}

Give 1234, 123, 12345 as input and see the result.

iv) #include<stdio.h>

main()

{

double x=5000.0,y=0.0025;

printf("%f %f %f %f\n\n" ,x,y,x\*y,x/y);

printf("%e %e %e %e\n", x,y,x\*y,x/y);

}

v) /\* minimum field width specification \*/

#include<stdio.h>

main()

{

int i = 12345;

float x = 345.678;

printf("%3d%5d%8d\n\n", i,i,i);

printf("%3g%15g%10g\n", x,x,x);

}

vi) /\*display a floating number with several precisions \*/

#include<stdio.h>

main()

{

float x = 123.456;

printf("%7f%7.3f%7.1f\n\n", x,x,x);

printf("%12e%12.5e%12.3e\n", x,x,x);

}

23 . Write a program to swap two numbers using a temporary variable.

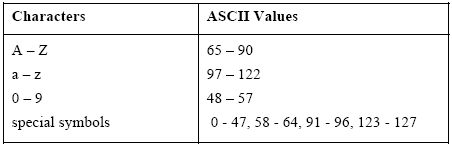
24 Write a program to calculate xn (without using the library function *pow(x,n)*).

25 Write a program to calculate the absolute value of an integer.

26 Any year is input through the keyboard. Write a program to determine whether the year is leap year or not.

27 Accept five long integer values from the user and find the average of it and display it in floating point and exponential notation.

28 Any character is entered through the keyboard; write a program to determine whether the character entered is a capital letter, a small case letter, a digit or a special symbol. The following table shows the range of **ASCII values for various characters.**



29 Try the following programs to understand printf() & scanf()

1. /\* This program illustrates the usage of various C datatypes \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include<stdio.h>

int main()

{

int a = 3000; /\* positive integer data type

float b = 4.5345; /\*float data type

char c = 'A'; /\*char data type

long d = 31456; /\* long positive integer data type

long e = -31456; /\* long -ve integer data type

int f = -145; /\* -ve integer data type

unsigned int fu = 145; /\* unsigned integer data type

short g = 120; /\* short integer data type

short h = -120; /\* short negative integer data type

double i = 5.1234567890; /\* double float data type

float j = -3.24; /\*float data type

printf("\nWelcome Students!!!\n");

printf("\nHere are the list of the C data types\n");

printf("\n1. This is positive integer number (int):\t\t%d", a);

printf("\n2. This is positive float number (float):\t\t%f", b);

printf("\n3. This is negative float number(float):\t\t%f", j);

printf("\n4. This is character data (char):\t\t\t%c", c);

printf("\n5. This is long positive integer number(long):\t\t %ld", d);

printf("\n6. This is long negative integer number(long):\t\t %ld", e);

printf("\n7. This is negative integer number(int):\t\t %d", f);

printf("\n8. This is positive integer number(unsigned int):\t %u", fu);

printf("\n9. This is short positive integer number(short):\t %hd", g);

printf("\n10. This is short negative integer number(short):\t %hd", h);

printf("\n11. This is double positive float number(double):\t %lf", i);

printf("\n\t---do you understand?----\n ");

return 0;

}

1. /\* Program to understand type casting \*/

/\*Typecasting the character value to the numeric representation of the character in \*/

/\*ASCII and using Escape Sequence to print single quotes \*/

#include<stdio.h>

int main()

{

int a=5 , b=4, c=10;

float average;

char ch='a';

average = (float) (a+b+c)/3;

printf("\nThe Average of 3 numbers 5,4 and 10 is %f",average);

printf("\nThe ASCII value of \'a\' is %d",(int) ch);

}

1. /\* Float format \*/

#include<stdio.h>

int main()

{

float n1;

printf("\n Enter a float value: ");

scanf("%f" ,&n1);

/\*Print float with 6 digits precision to the right of the decimal point by default \*/

printf("\n%f",n1);

/\*Printing float 3 integer numbers with only 2 digits(rounding-off) after decimal

value \*/

printf("%\n3.2f ", n1);

/\* Scientific notation of a floating point value in the exponential form with 6 digits

of precision \*/

printf("\n%e", n1);

/\* Scientific notation of a floating point value in the exponential form by rounding-

off to 2 digits of precision \*/

printf("\n%.2e", n1);

}

1. /\* Integer Formatting \*/

#include <stdio.h>

int main()

{

printf("Various format for integer printing\n");

printf("-------------------------------------\n");

printf("%d\n", 455);

printf("%i\n", 455); /\* i same as d in printf() \*/

printf("%d\n", +455);

printf("%d\n", -455);

printf("%hd\n", 32000);

printf("%ld\n", 2000000000L);

printf("%o\n", 455);

printf("%u\n", 455);

printf("%u\n", -455);

/\* -455 is read by %u and converted to the unsigned value 4294966841 by 4 bytes integer \*/

printf("%x\n", 455);

printf("%X\n", 455);

return 0;

}

1. //\* Format Justification \*/

#include <stdio.h>

int main()

{

printf("Right justifying and left justifying values.\n");

printf(" Compare the output with the source code.\n");

printf("--------------------------------------------\n\n");

printf("%10s%10d%10c%10f\n\n", "hello", 7, 'a', 1.23); /\* Right Justification \*/

printf("%-10s%-10d%-10c%-10f\n", "hello", 7, 'a', 1.23); /\* Left Justification \*/

return 0;

}

1. /\* Discard unnecessary Characters \*/

#include <stdio.h>

int main()

{

int month1, day1, year1, month2, day2, year2;

printf(" Enter a date in the form mm/dd/yy: ");

/\*pad 02 for two fields and discarding the / characters.... \*/

scanf("%d%\*c%d%\*c%d", &month1, &day1, &year1);

printf("month = %02d day = %02d year = %02d\n\n", month1, day1, year1);

return 0;

}

\*\*\*\*\*\*\*\*\*