Computer Assignment

Q1
```
#include <stdio.h>

int main() {
   float price, tax_rate, total_price;

   printf("Enter the price of the product: ");
   scanf("%f", &price);

   printf("Enter the sales tax rate: ");
   scanf("%f", &tax_rate);

   total_price = price + (price * tax_rate / 100);

   printf("The total price of the product after adding sales tax is: $%.2f\n", total_price);

   return 0;
}
```

Q2
```
#include <stdio.h>

int main() {
   float hours_worked, hourly_wage, weekly_wages;

   printf("Enter the number of hours worked: ");
   scanf("%f", &hours_worked);

   printf("Enter the hourly wage: ");
   scanf("%f", &hourly_wage);

   if (hours_worked > 30) {
       weekly_wages = (30 * hourly_wage) + ((hours_worked - 30) * (2 * hourly_wage));
   } else {
       weekly_wages = hours_worked * hourly_wage;
   }

   printf("The weekly wages of the employee is: $%.2f\n", weekly_wages);

   return 0;
}
```

```
Q3. #include <stdio.h>

int main() {
    float apple_price = 50.0, mango_price = 35.0, potato_price = 10.0, tomato_price = 15.0;
    float apple_weight = 2.0, mango_weight = 1.5, potato_weight = 2.5, tomato_weight = 1.0;
    float total_cost, amount_paid, amount_returned;

    total_cost = (apple_price * apple_weight) + (mango_price * mango_weight) + (potato_price * potato_weight) + (tomato_price * tomato_weight);
    amount_paid = 500.0;
    amount_returned = amount_paid - total_cost;

    printf("The amount shopkeeper will return to Mr. X is: Rs %.2f\n", amount_returned);

    return 0;
}
```

```
Q4. #include <stdio.h>

int main() {
    printf("Name: Aditya Saxena\n");
    printf("Date of Birth: November 13, 2005\n");
    printf("Mobile Number: 123-456-7890\n");
    return 0;
}
Q5. #include <stdio.h>

int main() {
    int integer_value;
    char character_value;
    float float_value;

    printf("Enter an integer: ");
    scanf("%d", &integer_value);

    printf("Enter a character: ");
    scanf(" %c", &character_value);

    printf("Enter a float value: ");
    scanf("%f", &float_value);

    printf("The integer value you entered is: %d\n", integer_value);
    printf("The character you entered is: %c\n", character_value);
    printf("The float value you entered is: %.2f\n", float_value);

    return 0;
}
Q6. #include <stdio.h>

int main() {
    float cost = 172.53;

    printf("The sales total is : $ %.2f\n", cost);
    return 0;
}
```

```
Q7. #include <stdio.h>

int main() {
    int apples_per_person = 6.5;
    int number_of_people = 3;
    float total_apples = apples_per_person * number_of_people;

    printf("Raju has a total of %.1f apples.\n", total_apples);

    return 0;
}
```

```
Q8. #include <stdio.h>

int main() {
    float value = 123.456789;
    printf("The value in exponential format is: %.2e\n", value);

    return 0;
}
```

```
Q9. #include <stdio.h>

int main() {
    long long int mobile_number;

    printf("Enter your mobile number: ");
    scanf("%lld", &mobile_number);

    printf("Your mobile number is: %lld\n", mobile_number);
    return 0;
}
Q10. #include <stdio.h>

int main() {
    int population = 30000;

    population += population * 0.2;
    population += population * 0.3;

    printf("The population after two years is: %d\n", population);

    return 0;
}
Q11. #include <stdio.h>

int main() {
    char c;

    printf("Enter a character: ");
    scanf("%c", &c);

    printf("The ASCII value of %c is %d.\n", c, c);

    return 0;
}
Q12. #include <stdio.h>

int main() {
    float basic_pay, HRA, TA, salary;
    printf("Enter the basic pay of the employee: ");
    scanf("%f", &basic_pay);

    HRA = 0.15 * basic_pay;
    TA = 0.2 * basic_pay;

    salary = basic_pay + HRA + TA;

    printf("The salary of the employee is: $%.2f\n", salary);

    return 0;
}
Q13. #include <stdio.h>
#include <math.h>

int main() {
    float xp, yp, xq, yq;
    float slope, angle;

    printf("Enter coordinates of point P (xp yp): ");
    scanf("%f %f", &xp, &yp);

    printf("Enter coordinates of point Q (xq yq): ");
    scanf("%f %f", &xq, &yq);

    slope = (yq - yp) / (xq - xp);
    angle = atan(slope) * 180 / M_PI;

    printf("The slope of the line passing through P and Q is: %.2f\n", slope);
    printf("The angle of inclination of the line passing through P and Q is: %.2f degrees\n", angle);
    return 0;
}
Q14. #include <stdio.h>
#include <math.h>

int main() {
    float xp, yp, xq, yq;
    float slope, angle;

    printf("Enter coordinates of point P (xp yp): ");
    scanf("%f %f", &xp, &yp);

    printf("Enter coordinates of point Q (xq yq): ");
    scanf("%f %f", &xq, &yq);

    slope = (yq - yp) / (xq - xp);
    angle = atan(slope) * 180 / M_PI;

    printf("The slope of the line passing through P and Q is: %.2f\n", slope);
    printf("The angle of inclination of the line passing through P and Q is: %.2f degrees\n", angle);

    return 0;
}
Q15. #include <stdio.h>

int main() {
    float wavelength, speed, frequency;

    printf("Enter the wavelength of the wave: ");
    scanf("%f", &wavelength);
    printf("Enter the speed of the wave: ");
    scanf("%f", &speed);

    frequency = speed / wavelength;

    printf("The frequency of the wave is: %.2f Hz\n", frequency);

    return 0;
}
Q16#include <stdio.h>
#include <math.h>

int main() {
    float initial_velocity = 30.0;
    float acceleration = 5.0;
    float distance = 70.0;
    float final_velocity;

    final_velocity = sqrt(pow(initial_velocity, 2) + 2 * acceleration * distance);

    printf("The final velocity of the car is: %.2f m/s\n", final_velocity);

    return 0;
}
Q17 #include <stdio.h>

int main() {
    float initial_velocity = 0.0;
    float acceleration = 4.0;
    float time = 3.0;
    float final_velocity, distance;

    final_velocity = initial_velocity + acceleration * time;
    distance = initial_velocity * time + 0.5 * acceleration * time * time;

    printf("The final velocity of the horse is: %.2f m/s\n", final_velocity);
    printf("The distance travelled by the horse is: %.2f m\n", distance);

    return 0;
}
Q18 #include <stdio.h>

int main() {
    int roll_number = 1915001;
    int last_four_digits_sum;

    last_four_digits_sum = (roll_number % 10000) / 1 + (roll_number % 100000) / 10000 + (roll_number % 1000000) / 100000 + (roll_number % 10000000) / 1000000;

    printf("The sum of last four digits of your university roll number is: %d\n", last_four_digits_sum);

    return 0;
}
Q19#include <stdio.h>

int main() {
    const float CM_TO_INCHES = 0.393701;
    const float KG_TO_POUNDS = 2.20462;

    int height_cm = 170;
    float weight_kg = 70.5;
    float height_feet = height_cm * CM_TO_INCHES / 12;

    float weight_pounds = weight_kg * KG_TO_POUNDS;


    printf("Height: %.2f feet\n", height_feet);

    printf("Weight: %.2f pounds\n", weight_pounds);


    return 0;

}
```

Q20 a) A character variable named option:

```
char option;
```

Copy

b) An integer variable sum initialized to 0:

```
int sum = 0;
```

Copy

c) A floating point variable product initialized to 1:

```
float product = 1.0;
```

```
Q21 #include <stdio.h>


int main() {

    int numbers[9];


    printf("Enter nine integers:\n");


    for (int i = 0; i < 9; i++) {

        scanf("%d", &numbers[i]);

    }


    printf("The numbers you entered are:\n");


    for (int i = 0; i < 9; i++) {

        printf("%d", numbers[i]);
        if ((i + 1) % 3 == 0) {
            printf("\n");
        } else {
            printf(", ");
        }
    }

    return 0;
}
```

Q22. Header files in C programming contain function prototypes, type definitions, macros, and other declarations that are used by multiple source files. They allow us to reuse code across multiple files without having to redefine everything from scratch. Header files are included using the #include preprocessor directive at the beginning of a C source file.

Header files have several uses in C programming:

They provide function prototypes that allow us to use functions defined in other source files.

They define constants, macros, and type definitions that are used throughout a program.

They provide access to standard library functions and types.

They allow us to organize our code into separate modules for better maintainability.

Q23. The output of the following program will be:

```
56      70      38
```

Q24. The output of the following program will be:

```
GLA UNIVERSITY15
```

Q25. Library functions in C programming are pre-defined functions that are provided by standard libraries such as <stdio.h>, <stdlib.h>, <math.h>, etc. They provide commonly used functionality that can be reused across multiple programs without having to rewrite everything from scratch.

Here are four examples of library functions:

printf: Used for formatted output.

scanf: Used for formatted input.

sqrt: Used for calculating square root.

`rand

Q26. The output of the following program will be:

C is placement oriented LanguageHi 0 0

Q27. The statement printf("%d",scanf("%d%d",&a,&b)); reads two integers from standard input and stores them in variables a and b. It then returns the number of items successfully read from standard input, which is 2 in this case. Finally, it prints out the return value using printf function.

Q28. The output of the following program will be:

 "C % FOR % PLACEMENT"

Q29. #include <stdio.h>

```
int main() {

    float distance_km;

    float time_hrs = 4;

    float speed_kmph;
    printf("Enter distance between GLA University and Delhi (in km): ");
    scanf("%f", &distance_km);

    speed_kmph = distance_km / time_hrs;

    printf("Speed of bus: %.2f km/h\n", speed_kmph);

    return 0;
}
Q30. #include <stdio.h>

int main() {
    int satyam_marks = 50;
    int suman_marks = 70;
    int shyam_marks = 80;
    float average_marks;

    average_marks = (satyam_marks + suman_marks + shyam_marks) / 3.0;

    printf("Average marks: %.2f\n", average_marks);

    return 0;
}
Q31. #include <stdio.h>
```

```
int main() {
    int saurav_money = 100;
    int sajal_money = 200;
    int temp;

    printf("Before rectification:\n");
    printf("Saurav has %d rupees.\n", saurav_money);
    printf("Sajal has %d rupees.\n", sajal_money);

    temp = saurav_money;
    saurav_money = sajal_money;
    sajal_money = temp;

    printf("After rectification:\n");
    printf("Saurav has %d rupees.\n", saurav_money);
    printf("Sajal has %d rupees.\n", sajal_money);

    return 0;
}
Q32#include <stdio.h>

int main() {
    float speed_kmph = 4.0;
    float time_min = 3.0;
    float distance_km;
    distance_km = speed_kmph * time_min / 60;



    printf("Distance travelled: %.2f km\n", distance_km);



    return 0;

}
```

Q33. Yes, two or more escape sequences such as \n and \t can be combined in a single line of program code.

Q34. Comments in C programming are used to provide additional information about the code to make it more readable and understandable. Comments are ignored by the compiler and do not affect the execution of the program. There are two types of comments in C programming:

Single-line comments: These comments start with // and continue until the end of the line.

Multi-line comments: These comments start with /* and end with */. They can span multiple lines.

Q35. The statement is missing an ampersand (&) before the variable name. The correct statement should be:

```
scanf("%d", &number);
```

Q36. The output of the following program will be:

```
Yes
```

The program uses an if-else statement to check if the size of an integer is greater than -1. Since the size of an integer is always greater than -1, the condition is true, and the program prints out "Yes" using printf function.

Q37. The following variable names are invalid:

gross-salary (contains a hyphen)

avg. (contains a period)

thereisbookinmysoup (no spaces between words)

```
Q38. #include <stdio.h>

int main() {
    float tank_size_gallons = 175;
    float drain_rate_gph = 25;
    float time_hours;

    time_hours = tank_size_gallons / drain_rate_gph;

    printf("Time required to completely clean tank: %.2f hours\n", time_hours);

    return 0;
}
```

```
Q39. #include <stdio.h>

int main() {
    float battery_power_percent = 75;
    float battery_power_remaining_percent = -0.2 * x + 1;
    printf("Battery power remaining at %.0f%% after %.2f hours

}
```

Q40. The correct answer is a. Compiler.

Q41. The format specifier for an octal number is %o.

Q42. The format specifier used to print the exponent value up to 2 decimal places is %.2e.

Q43. The correct answer is b. array.

Q44. The output of the following code will be hello7.

Q45. The output of the following code will be Garbage, 5.

Q46. The correct answer is b. Basic_pay.

Q47. The output of the following program will be c1.

Q48. Here are the answers to your questions: a) (365.55)10 = (101101101.1000110011)2 b) (453.65)10 = (1655.52)8 c) (5164.12)10 = (141C.1EB851EB852)16 d) (23.65)10 = (113.2)5 e) (772)10 = (2203)7

Q49. Here are the answers to your questions: a) (325.54)6 = (755.4)10 b) (1001010110101.1110101)2 = (4781.84375)10 c) (742.72)8 = (386.578125)10 d) (AC94.C5)16 = (44244.7724609375)10

Q50. Here are the answers to your questions: (DB56.CD4)16 =

In binary: 1101101101010110.11001101000100

In octal: 6652566.6222

In decimal: 144240.7998046875

Q51. Here are the answers to your questions: (473.42)8 =

In binary: 100111001.100010

In decimal: 315.25

In hexadecimal: 13D.C

Q52. a) A=11 b) A=5 c) A=3

Q53: The output of the following program will be -32766.

The variable a is declared as an integer and initialized with a value of 32770, which is greater than the maximum value that can be stored in a signed 2-byte integer variable, which is 32767. Therefore, the value of a will overflow and become -32766 due to two’s complement representation of signed integers on most systems.

Q54: The output of the following program will be Temperature in Fahrenheit is 41.00.

(Aditya Saxena)