CS ASSIGNMENT

NAME :NIKHIL DEY

CLASS : A-Y

ROLL NO : 33

SUBMITTED TO :

MRS. GURPREET KAUR

Q1. Write a C program for calculating the price of a product after adding the sales tax to its original price. Where rate of tax and price is inputted by user.

```
#include <stdio.h>

int main() {
    float taxRate, originalPrice;

    printf("Enter the tax rate (in percentage): ");
    scanf("%f", &taxRate);

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

    float taxAmount = (taxRate / 100) * originalPrice;
    float totalPrice = originalPrice + taxAmount;

    printf("Original Price: $%.2f\n", originalPrice);
    printf("Tax Amount (%.2f%%): $%.2f\n", taxRate, taxAmount);
    printf("Total Price: $%.2f\n", totalPrice);

    return 0;
}
```

Q2. Write a C program to calculate the weekly wages of an employee. The pay depends on wages per hour and number of hours worked. Moreover, if the employee has worked for more than 30 hours, then he or she gets twice the wages per hour, for every extra hour that he or she has worked.

```
#include <stdio.h>
int main() {
    float wagesPerHour, hoursWorked, weeklyWages;

    printf("Enter the wages per hour: ");
    scanf("%f", &wagesPerHour);

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

    if (hoursWorked <= 30) {
        weeklyWages = wagesPerHour * hoursWorked;
    } else {
        weeklyWages = (30 * wagesPerHour) + ((hoursWorked - 30) * (2 * wagesPerHour));
    }

    printf("Weekly Wages: $%.2f\n", weeklyWages);

    return 0;
}
```

Q.3 Mr. X goes to market for buying some fruits and vegetables. He is having a currency of Rs 500 with him for marketing. From a shop, he purchases 2.0 kg Apple priced Rs. 50.0 per kg, 1.5 kg Mango priced Rs.35.0 per kg, 2.5 kg Potato priced Rs.10.0 per kg, and 1.0 kg Tomato priced Rs.15 per kg. He gives the currency of Rs. 500 to the shopkeeper. Find out the amount shopkeeper will return to X by writing a C program.

```
#include <stdio.h>
int main() {
    float totalCost = (2.0 * 50.0) + (1.5 * 35.0) + (2.5 * 10.0) + (1.0 * 15.0);
    float payment = 500.0;
    float change = payment - totalCost;

    printf("Amount to be returned to Mr. X: Rs. %.2f\n", change);

    return 0;
}
```

Q4.Write a C program to print your name, date of birth and mobile number in 3 different lines.

```
#include <stdio.h>

int main() {
    printf("Your Name\n");
    printf("Date of Birth\n");
    printf("Mobile Number\n");

    return 0;
}
```

Q5.Write a program to read an integer, a character and a float value from keyboard and display the same in different lines on the screen.

```
#include <stdio.h>

int main() {
    int integerNumber;
    char character;
    float floatNumber;

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

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

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

    printf("You entered:\n");
    printf("%d\n", integerNumber);
    printf("%c\n", character);
    printf("%.2f\n", floatNumber);

    return 0;
}
```

Q6.Write a program to print the following line (Assume the total value is contained in a variable named cost)

```
#include <stdio.h>

int main() {
   float cost = 172.53;
   printf("The sales total is : $ %.2f\n", cost);
   return 0;
}
```

The sales total is : $ 172.53

Q7.Raju got 6 and half apples from each of Raghu, Sheenu and Akash. He wants to know how many apples he has in total without adding them. Write a program which could help Raju in doing this.

```
#include <stdio.h>

int main() {
   int applesFromRaghu = 6;
   int applesFromSheenu = 6;
   int applesFromAkash = 6;

   int totalApples = applesFromRaghu + applesFromSheenu + applesFromAkash;

   printf("Raju has a total of %d apples.\n", totalApples);
```

```
    return 0;
}
```

Q8.Write a program that prints the floating point value in exponential format correct to two decimal places.

```
#include <stdio.h>

int main() {
    float floatValue = 1234.56789;
    printf("%.2e\n", floatValue);

    return 0;
}
```

Q9.Write a program to input and print your mobile number (i.e. of 10 digits).

```
#include <stdio.h>

int main() {
    long long int mobileNumber;

    printf("Enter your 10-digit mobile number: ");
    scanf("%lld", &mobileNumber);

    printf("Your mobile number is: %lld\n", mobileNumber);
```

```
    return 0;
}
```

Q10.The population of a city is 30000. It increases by 20 % during first year and 30% during the second year. Write a program to find the population after two years? (Ans: 46800)

```
#include <stdio.h>

int main() {
    int population = 30000;

    population = population + (0.20 * population);
    population = population + (0.30 * population);

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

    return 0;
}
```

Q11. Write a program to find the ASCII value of a character.

```
#include <stdio.h>

int main() {
    char ch;

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

```
    printf("ASCII value of %c is %d\n", ch, ch);

    return 0;
}
```

Q12. Write a program to calculate salary of an employee, given his basic pay (entered by user), HRA=15% of the basic pay and TA=20% of the basic pay.

```
#include <stdio.h>

int main() {
    float basicPay, HRA, TA, salary;

    printf("Enter the basic pay: ");
    scanf("%f", &basicPay);

    HRA = 0.15 * basicPay;
    TA = 0.20 * basicPay;

    salary = basicPay + HRA + TA;

    printf("Salary: %.2f\n", salary);

    return 0;
}
```

Q13. Write a program to find the slope of a line and angle of inclination that passes through two points P and Q with coordinates (xp, yp) and (xq, yq) respectively.

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

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

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

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

    slope = (yq - yp) / (xq - xp);
    angle = atan(slope) * (180.0 / M_PI);

    printf("Slope: %.2f\n", slope);
    printf("Angle of Inclination: %.2f degrees\n", angle);

    return 0;
}
```

Q14. The SPI (Semester Performance Index) is a weighted average of the grade points earned by a student in all the courses he registered for in a semester. If the grade points associated with the letter grades awarded to a student are g1, g2, g3,…….gk etc. and the corresponding credits are c1, c2, c3,……ck, the SPI is given by:

$$\text{SPI} = \frac{\sum_{i=1}^{k} c_i g_i}{\sum_{i=1}^{k} c_i}$$

Where, k is the number of courses for which the candidate remains registered for during the semester/ trimester. Write a program in C to calculate SPI for k =5.

Q 15. Write a program to calculate the frequency (f) of a given wave with wavelength (λ) and speed (c), where c=λ*f.

```
#include <stdio.h>

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

    printf("Enter the wavelength (λ): ");
    scanf("%f", &wavelength);

    printf("Enter the speed (c): ");
    scanf("%f", &speed);

    frequency = speed / wavelength;

    printf("Frequency (f): %.2f\n", frequency);

    return 0;
}
```

Q 16. A car travelling at 30 m/s accelerates steadily at 5 m/s2 for a distance of 70 m. What is the final velocity of the car? [Hint: v2 = u2 + 2as]

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

```
int main() {
    float initialVelocity = 30.0;
    float acceleration = 5.0;
    float distance = 70.0;

    float finalVelocity = sqrt(initialVelocity * initialVelocity + 2 * acceleration * distance);

    printf("Final velocity of the car: %.2f m/s\n", finalVelocity);

    return 0;
}
```

Q 17.A horse accelerates steadily from rest at 4 m/s2 for 3s. (a) What is its final velocity? (b) How far has it travelled? [Hint: (a) v = u + at (b) s = ut + ½at2]

```
#include <stdio.h>

int main() {
    float initialVelocity = 0.0;
    float acceleration = 4.0;
    float time = 3.0;

    float finalVelocity = initialVelocity + (acceleration * time);
    float distance = (initialVelocity * time) + (0.5 * acceleration * time * time);

    printf("(a) Final velocity: %.2f m/s\n", finalVelocity);
    printf("(b) Distance traveled: %.2f meters\n", distance);
```

```
    return 0;
}
```

Q 18. Write a program to find the sum of your four last digit of your university roll number .

```
#include <stdio.h>

int main() {
    char rollNumber[] = "YOUR_ROLL_NUMBER";

    int sum = 0;
    int i = 0;

    for (i = strlen(rollNumber) - 4; i < strlen(rollNumber); i++) {
        if (isdigit(rollNumber[i])) {
            sum += (rollNumber[i] - '0');
        }
    }

    printf("Sum of the last four digits of your roll number: %d\n", sum);

    return 0;
}
```

Q19. Write a program to initialize your height and weight in cm. and kgs respectively demonstrating compile time initialization and convert them in feets and pounds respectively. Note :- 1 cm = 0.393701inch , 1 Kg = 2.20462

```
#include <stdio.h>


int main() {
    const float cmToInch = 0.393701;
    const float kgToPound = 2.20462;


    float heightInCm = 180.0;
    float weightInKg = 75.0;


    float heightInInch = heightInCm * cmToInch;
    float weightInPound = weightInKg * kgToPound;


    printf("Height: %.2f feet\n", heightInInch / 12);
    printf("Weight: %.2f pounds\n", weightInPound);


    return 0;
}
```

Q 20 . Code the variable declarations for each of following:

a) A character variable named option.
b) An integer variable sum initialized to 0
c) A floating point variable, product, initialized to 1

a)char option;

b)int sum = 0;

c)float product = 1;

Q21. Write a program that reads nine integers. Display these numbers by printing three numbers in a line separated by commas.

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

```
int main() {
    int numbers[9];
    printf("Enter nine integers:\n");

    // Read nine integers
    for (int i = 0; i < 9; i++) {
        scanf("%d", &numbers[i]);
    }

    // Display the numbers in groups of three
    for (int i = 0; i < 9; i++) {
        printf("%d", numbers[i]);
        if ((i + 1) % 3 == 0) {
            printf("\n");
        } else {
            printf(", ");
        }
    }

    return 0;
}
```

Q22. What are header files and what are its uses in C programming?

Header files in C programming are files that contain declarations and sometimes definitions for functions, variables, and other constructs used in a program. They serve several important purposes:

1. **Modularization:** Header files allow you to divide your code into manageable modules. You can declare functions and data structures in header files and implement them in separate source files. This promotes code organization and makes it easier to maintain and collaborate on larger programs.

2. **Reusability:** Header files facilitate code reuse. You can include a header file in multiple source files, enabling you to use the same functions and data structures across different parts of your program without duplicating code.

3. **Encapsulation:** Header files provide an interface to your code. They allow you to hide the implementation details of functions and data structures from users of your code. Users only need to know the function prototypes and how to use them, not the internal workings.

4. **Type Safety:** Header files help enforce type safety by providing function prototypes. If you attempt to call a function with the wrong number or type of arguments, the compiler will issue an error, helping catch bugs early.

5. **Standardization:** C standard library functions are typically declared in header files. By including these headers, you gain access to a wide range of pre-built functions and data types.

Commonly used C header files include `<stdio.h>` for input and output, `<stdlib.h>` for memory allocation and other utilities, `<string.h>` for string manipulation, and `<math.h>` for mathematical functions.

In summary, header files in C are essential for organizing, reusing, encapsulating, and ensuring type safety in your code. They are a fundamental part of C programming, enabling you to build complex and modular applications efficiently.

Q23. What will be the output of following program?

```
#include<stdio.h>
int main()
{   int num=070;
printf("%d\t%o\t%x",num,num,num);
}
```

56 70 38

Q 24. What will be the output of following program?

```
#include <stdio.h>
void main()

{

int x = printf("GLA UNIVERSITY");

printf("%d", x);

 }
```

GLA UNIVERSITY14

Q25. What are library functions? List any four library functions.

Library functions in programming refer to predefined functions that are provided as part of a programming language's standard library or additional libraries. These functions are written, tested, and optimized by experienced programmers to perform common and often complex tasks. Programmers can use these functions to save time and effort in their own code by calling them instead of implementing the functionality from scratch. Library functions cover a wide range of tasks, including input/output operations, mathematical calculations, string manipulation, memory management, and more.

Here are four common library functions in the C programming language:

1. **printf()**: This function is part of the standard I/O library (`<stdio.h>`) and is used to print formatted output to the console.

Example:

```c
printf("Hello, World!\n");
```

2. **strlen()**: This function, from the string manipulation library (`<string.h>`), calculates the length of a null-terminated string.

Example:

```c
#include <string.h>
char str[] = "Hello";
int length = strlen(str); // length will be 5
```

3. **sqrt()**: This function, from the math library (`<math.h>`), calculates the square root of a number.

Example:

```c
#include <math.h>
double result = sqrt(16.0); // result will be 4.0
```

4. **malloc()**: This function, from the memory allocation library (`<stdlib.h>`), is used to allocate dynamic memory on the heap.

Example:

```c
```

```
#include <stdlib.h>

int *arr = (int *)malloc(5 * sizeof(int));
```

These are just a few examples of library functions in C. Many more functions are available in C's standard library and additional libraries, providing a wide range of functionality for developers to use in their programs.

Q26. What will be the output of following program?

```
#include <stdio.h>
void main()
{
  int x = printf("C is placement oriented Language") – printf(“Hi”);
  printf("%d %o %x", x,x,x);
}
```

31 37 1f

Q27. What is the meaning of following statement? printf(“%d”,scanf(“%d%d”,&a,&b));

The given statement `printf("%d", scanf("%d%d", &a, &b));` reads two integer values from the user and stores them in variables `a` and `b`. It then prints the number of successful inputs (usually 2) to the console.

In simpler terms, it tells you how many integers you successfully entered when prompted.

Q28. What will be the output of following program?

```
#include <stdio.h>
void main()
{
  printf(" \"C %% FOR %% PLACEMENT\"");
}
```

Q29. Suppose distance between GLA University and Delhi is m km (to be entered by user), by BUS you can reach Delhi in 4 hours. Develop a 'C' program to calculate speed of bus.

```
#include <stdio.h>


int main() {
    float distance, time, speed;


    // Input distance from the user
    printf("Enter the distance (in kilometers): ");
    scanf("%f", &distance);


    // Time to reach Delhi by bus (4 hours)
    time = 4.0;  // hours


    // Calculate the speed (Speed = Distance / Time)
    speed = distance / time;


    // Display the calculated speed
    printf("The speed of the bus is %.2f km/h\n", speed);


    return 0;
}
```

Q30. In an exam Satyam got 50 marks, Suman got 70 marks and Shyam got 80 marks, Write a 'C' program to find average marks of these three participants.

```
#include <stdio.h>

int main() {
    int satyamMarks = 50;
    int sumanMarks = 70;
    int shyamMarks = 80;
    int totalParticipants = 3;
    float averageMarks;

    int totalMarks = satyamMarks + sumanMarks + shyamMarks;
    averageMarks = (float)totalMarks / totalParticipants;

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

    return 0;
}
```

Q31. One day, Mohan called Saurav and Sajal and gave some money to them, later he realized that money that was given to Saurav should be given to Sajal and vice-versa. Develop a 'C' program to help Mohan so that he can rectify his mistake.

```
#include <stdio.h>

int main() {
    float sauravMoney, sajalMoney, temp;

    printf("Enter the amount of money given to Saurav: ");
    scanf("%f", &sauravMoney);
```

```
    printf("Enter the amount of money given to Sajal: ");
    scanf("%f", &sajalMoney);

    temp = sauravMoney;
    sauravMoney = sajalMoney;
    sajalMoney = temp;

    printf("After rectifying the mistake:\n");
    printf("Amount of money given to Saurav: %.2f\n", sauravMoney);
    printf("Amount of money given to Sajal: %.2f\n", sajalMoney);

    return 0;
}
```

Q32. One day when I was going for a lunch, suddenly rain started, I was very hungry so started running with speed of 4km/h and it took 3 min to reach mess. Help me to develop a 'C' program to calculate distance travelled by me.

```
#include <stdio.h>

int main() {
    float speed = 4.0;
    float timeInMinutes = 3.0;
    float timeInHours, distance;

    timeInHours = timeInMinutes / 60.0;
    distance = speed * timeInHours;
```

```
    printf("Distance traveled: %.2f kilometers\n", distance);

    return 0;
}
```

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

Yes, you can combine multiple escape sequences in a single line of program code in C or many other programming languages. For example, you can use both `\n` (newline) and `\t` (tab) in a single line of code to format output or create specific line breaks and indentation. Here's an example:

```c
#include <stdio.h>

int main() {
    printf("Hello,\n\tWorld!\n");
    return 0;
}
```

In this code, both `\n` and `\t` are used within the double-quoted string to format the output with a newline and a tab, resulting in the following output:

```
Hello,
    World!
```

```

So, it's perfectly valid to use multiple escape sequences in a single line of code to achieve the desired formatting or behavior.

Q34. What are comments and how do you insert it in a C program?

Comments in a C program are non-executable statements that are used to provide explanations, descriptions, or notes within the source code. Comments are ignored by the C compiler and are meant for human readers to understand the code better. They are crucial for documenting code, making it more readable, and helping other programmers (including your future self) understand your code.

In C, there are two common ways to insert comments:

1. **Single-Line Comments:** Single-line comments are used for brief comments that span only one line. They start with `//` and continue until the end of the line. Here's an example:

   ```c
   // This is a single-line comment
   int x = 10; // This comment explains the variable initialization
   ```

2. **Multi-Line Comments:** Multi-line comments are used for longer comments that may span multiple lines. They start with `/*` and end with `*/`. Everything between `/*` and `*/` is considered a comment. Here's an example:

   ```c

```
/*
This is a multi-line comment.
It can span multiple lines.
Use it for more extensive explanations.
*/

int y = 20; /* This is a comment on the same line as code */
```

It's important to use comments to document your code, including explanations of complex algorithms, the purpose of functions and variables, and any known issues or limitations. Well-documented code is easier to understand, maintain, and debug.

Q35. What is wrong in this statement? scanf(“%d”,number);

The issue with the statement `scanf("%d", number);` is the use of incorrect double-quotes for the format specifier and the missing ampersand ('&') before the variable name.

In C, when using `scanf` to read a value into a variable, you should use double-quotes (`"`) for the format specifier, and you need to provide the memory location where you want to store the value using the address-of operator (`&`).

So, the correct statement should be:

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

This change ensures that `scanf` correctly reads an integer value and stores it in the memory location pointed to by `number`.

Q36. What will be the output?

```
#include <stdio.h>
    int main()
{
   if (sizeof(int) > -1)
      printf("Yes");
   else
      printf("No");
   return 0;
}
```

The output of this program will be "Yes."

Q37. Point out which of the following variable names are invalid:

gross-salary INTEREST , salary of emp , avg. , thereisbookinmysoup

Among the provided variable names, the invalid variable names are:

1. `gross-salary`: Variable names cannot contain a hyphen (-) in C.
2. `avg.`: Variable names cannot contain a period (.) in C.

The valid variable names are:

1. `INTEREST`
2. `salary_of_emp`
3. `thereisbookinmysoup`

Q38. Tom works at an aquarium shop on Saturdays. One Saturday, when Tom gets to work, he is asked to clean a 175-gallon reef tank. His first job is to drain the tank. He puts a hose into the tank and starts a siphon. Tom wonders if the tank will finish draining before he leaves work. He measures the amount of water that is draining out and finds that 12.5 gallons drain out in 30 minutes. So, he figures that the rate is 25 gallons per hour. Develop a 'C' program to help Tom to calculate time required to completely clean tank.

```
#include <stdio.h>

int main() {
    float tankVolume = 175.0;
    float drainageRate = 25.0;
    float timeRequired;

    timeRequired = tankVolume / drainageRate;

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

    return 0;
}
```

Q39. The percent y (in decimal form) of battery power remaining x hours after you turn on a laptop computer is y = −0.2 x + 1. Develop a 'C' program to calculate after how many hours the battery power is at 75%?

```
#include <stdio.h>

int main() {
    float batteryPower = 0.75;  // 75% battery power
    float hours;

    // Calculate the number of hours (x) when battery power is at 75%
    hours = (batteryPower - 1) / (-0.2);

    // Display the result
    printf("The battery power is at 75%% after %.2f hours.\n", hours);
```

```
    return 0;
}
```

Q40.Which of the following is used to convert the high level language in machine language in a single go?

a. Compiler b.Interpreter

c. Linker d.Assembler

a. Compiler

Q 41. What is the format specifier for an Octal Number?

a.%0 b.%d

c. %o d. %e

c. %o

Q 42. Which format specifier is used to print the exponent value upto 2 decimal places.

a. %e b.%.2f c. %f d.%.2e

d. %.2e

Q 43. Which of the following is not a basic data type?

a. char

b. array

c. float

d. int

b. array

Q 44. What is the output of following code?

```
#include<stdio.h>
void main()
```

```
{
  int x=0;
  x= printf("\"hello\b\"");
  printf("%d",x);
}
```

a. hello7 b. "hello"7 c. "hell"8 d. hell8

c. "hell"8

Q 45. What is the output of following code?

```
#include<stdio.h>
void main()
{
  int b,c=5 ;
  int("%d , %d", b,c);
}
```

a. 5, 5 b. 5, 5.000000

c. Garbage, 5.000000 d. Garbage, 5

d. Garbage, 5

Q46. Which of the following is an identifier?

a. &fact b. Basic_pay c. enum d. 1sum

b. Basic_pay

Q 47. What is the output of the following program?

```
#include<stdio.h>
void main()
{
   char x, a='c';
   x=printf("%c",a);
```

```
  printf("%d",x);
}
```

a. c1 b. cgarbage

c. 1 c. c

a. c1

Q48. Perform the following conversion from Decimal to other number as directed-

a) $(365.55)_{10} = (101101101.11001000)_2$
b) $(453.65)_{10} = (705.514)_8$
c) $(5164.12)_{10} = (14C1.1EB8)_{16}$
d) $(23.65)_{10} = (43.311)_5$
e) $(772)_{10} = (2152)_7$

Q49. Covert the following numbers to decimal number system-

a) $(325.54)_6 = (125 + 17/18)_{10}$
b) $(1001010110101.1110101)_2 = (4809.9140625)_{10}$
c) $(742.72)_8 = (482 + 29/32)_{10}$
d) $(AC94.C5)_{16} = (44180.76953125)_{10}$

Q50. Perform the following conversion from Hexadecimal to other number as directed-

$(DB56.CD4)_{16} = (11011011011010100110011010 0)_2$, $(6656514614)_8$, $(1311613414)_4$

Q51. Perform the following conversion from octal to other number as directed-

$(473.42)_8 = (10011100100.010010)_2$, $(315.53125)_{10}$, $(13B.8A3D70A3D70A3D7)_{16}$, $(120.3423)_5$

Q52. Find the value of A?

a) $(23)_{10} = (17)_{A=7}$
b) $(21)_{16} = (41_{)A=10}$
c) $(32)_8 = (101)_{A=33}$

Q53: What will be the output of following program? Assume integer is of 2 bytes

```
void main(){
int  a=32770;
printf(“%d”,a);
}
```

The output of this program is undefined

Q53: What will be the output of following program? Assume integer is of 2 bytes

```
void main(){
int  a=32770;
printf(“%d”,a);
}
```

-32766

Q54:
```
#include <stdio.h>
 int main()
{
 float c = 5.0;
 printf ("Temperature in Fahrenheit is %.2f", (9/5)*c + 32);
 return 0;
}
```

Temperature in Fahrenheit is 41.00