

Computer programming Assignment

Answer - 1

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

```
int main()
```

```
{ float price, tax Rate , total Price ;
```

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

```
printf("Enter the tax rate:");  
scanf("%f", & Rate & tax rate);
```

```
total Price = Price + (Price * tax Rate / 100);
```

```
printf("The total price after adding tax  
is: %f/n", total Price);
```

```
return 0;
```

```
}
```

Answer - 2

include < stdio.h >

int main()

{

 float wage_per_hour, hours_worked,
 weekly_wages;

 printf("Enter the wage per hour: ");
 scanf("%f", &wage_per_hour);

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

 if (hours_worked <= 30)

 { weekly_wages = wage_per_hour *

 hours_worked;

}

 else { float extra_hours = hours_worked - 30;

 weekly_wages = (wage_per_hour * 30) +

 (wage_per_hour * 2 * extra_hours);

}

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

 return 0;

}

Page

Answer - 3

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

```
int main () {
```

```
    float apple - price - per - kg = 50.0;
```

```
    float mango - price - per - kg = 35.0;
```

```
    float Potato - price - per - kg = 10.0;
```

```
    float tomato - price - per - kg = 15.0;
```

```
    float apple - weight = 2.0;
```

```
    float mango - weight = 1.5;
```

```
    float Potato - weight = 2.5;
```

```
    float tomato - weight = 1.0;
```

```
    float total - cast = (apple - price - per - kg *  
    apple - weight) + (mango - price - per - kg *  
    mango - weight) + (Potato - price - per - kg *  
    Potato - weight) + (tomato - price - per - kg *  
    tomato - weight);
```

```
    float currency = 500.0;
```

```
    float amount - returned = currency -  
    total cast;
```

```
    printf ("The amount the shopkeeper will return  
to Mr. X is: % .2f\n", amount - returned);
```

```
    return 0;
```

```
}
```

Answer-4

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

```
int main()
{
    printf (" Ayush Agarwal \n ");
    printf (" 26th August 2005 \n ");
    printf (" 6393592947 " );
    return 0;
}
```

Answer-5

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

```
int main()
{
    int A;
    int B;
    int C;
    scanf ("%d %c %f", &A &B &C);
    printf ("%d \n ", A);
    printf ("%c \n ", B);
    printf ("%f \n ", C);
    return 0;
}
```

Answer - 6

```
#include <stdio.h>
int main()
{
    float cast;
    cast = 172.53;
    printf ("The Sale total is : $ %.2f", cast);
    return 0;
}
```

Answer - 7

```
#include <stdio.h>
int main()
{
    float Apples;
    Apples = 6.5 * 3;
    printf ("%.2f", Apples);
    return 0;
}
```

Answer - 8

```
#include <stdio.h>
int main ()
{
    float number;
    scanf ("%f", &number);
    printf ("% .2f\n % .2f", number,
           number);
    return 0;
}
```

Answer - 9

```
#include <stdio.h>
int main ()
{
    long int male;
    scanf ("%d", &male);
    printf ("%ld", male);
    return 0;
}
```

Answer-10

```
#include < stdio.h>
#include < math.h>
int main()
{
    long int A, B;
    A = 30000 + (0.2 * 30000);
    B = A + (0.3 * A);
    printf ("%d", B);

    return 0;
}
```

Answer-11

```
#include < stdio.h>
int main()
{
    char n;
    scanf ("%c", &n);
    printf ("ASCII value is: %d", n);

    return 0;
}
```

Answer - 12

```
# include < stdio.h >
int main()
{
    float basic-pay, CTC;
    printf ("Enter value of basic-pay");
    scanf ("%f", & basic-pay);
    CTC = (basic-pay) + (basic-pay * 0.15) +
        (basic-pay * 0.20);
}
```

printf ("The CTC of this man is : %f",
 (CTC));

return 0;

Answer - 13

```
# include < stdio.h >
# include < Math.h >
int main ()
{
    int np, nq, yp, nq, yq;
    float Slope, angle;
    printf ("Enter coordinates of two points:");
    scanf ("%d %d %d %d", &np, &nq, &yp, &yq);
    Slope = (yp - yq) / (np - nq);
```

Printf ("slope = $y \cdot f \ln$ ", slope);

Angle = $a + \tan(\text{slope})$;

Printf ("angle = $y \cdot f \ln$ ", angle + $180 / 3.14$);

return 0;

}

Answer-14

include < stdio.h >

int main()

{

int g₁, g₂, g₃, g₄, g₅, c₁, c₂, c₃, c₄, c₅, SPI;

Printf ("Enter grades : \n");

Scanf ("%d %d %d %d %d", &g₁, &g₂, &g₃, &g₄, &g₅);

Printf ("Enter credit points : \n");

Scanf ("%d %d %d %d %d", &c₁, &c₂, &c₃, &c₄, &c₅);

SPI = $1.0 * ((c_1 * g_1) + (c_2 * g_2) + (c_3 * g_3) + (c_4 * g_4) + (c_5 * g_5)) / (c_1 + c_2 + c_3 + c_4 + c_5)$;

Printf ("SPI = %d", SPI);

return 0;

}

Page

Answer - 15

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

```
int main()
```

```
{
```

```
    float freq, Speed, WL;
```

```
    printf ("Enter Wavelength: ");
```

```
    scanf ("%f", &WL);
```

```
    printf ("Enter Speed");
```

```
    scanf ("%f", &Speed);
```

$$\text{freq} = \text{Speed} / \text{WL}$$

```
    printf ("freq of the wave: %f", freq);
```

```
    return 0;
```

```
}
```

Answer - 16

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

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

```
{
```

```
float v
```

```
int u, a, s;
```

```
a = 5;
```

```
u = 30;
```

```
s = 70;
```

Page _____

```
v = Pow ( u*u + 2*a*s , 0.5 );
printf ("The final velocity = %f " v);
return 0;
```

{

Answer-17

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

```
int main ()
```

{

```
float s;
```

```
int a = 4 , v , t=s , u=0;
```

$$v = u + a*t ;$$

```
printf ("%d \n , v);
```

$$s = (u*t) + (0.5 * a * Pow(t, 2));$$

```
printf ("%f " , s);
```

```
return 0;
```

{

Answer - 18

include < stdio.h >
int main()

{
long long int RN, a, b, c, d, S;
printf ("Give your overall no. = ");
scanf ("%lld", &RN);

a = RN % 10;

RN = RN / 10;

b = RN % 10;

RN = RN / 10;

c = RN % 10;

RN = RN / 10;

d = RN % 10;

S = a + b + c + d

printf ("The sum is : %d", S);

return 0;

}

Answer-19

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

```
int main()
```

```
{ float h, w, feet, Pound;
```

```
scanf ("%f %f", &h, &w);
```

```
feet = h * 0.393701;
```

```
Pound = w * 2.20462;
```

```
printf ("height in feet is %.2f weight in  
Pound is %.2f", feet, Pound);
```

```
return 0;
```

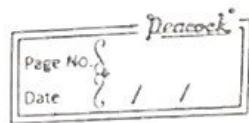
```
}
```

Answer-20

(a) char option

(b) int sum = 0;

(c) float product = 1.0;



Answer No - 21

In C programming, header files contain function prototypes and definitions that are used in multiple source files. They help in organizing and modularizing code by providing a way to declare functions, constants, and data structures. They are included using the '#include' directive to make these declarations available in other files.

Answer No - 23

The output of the given programme will be
56, 70, 30.

Answer no - 24

The output of the given programme will be
GILA UNIVERSITY14.

Explanation -

The printf function is used to print the string "GILA UNIVERSITY".

The printf function returns the number of characters printed, which is 14 in this case.

The value of 'x' is assigned the return value of printf.

Finally %d is used to print the value of 'x' which is 14.

Answer No - 25

In C programming, library functions are pre-defined functions that are provided by the C standard library. These functions are already implemented and can be used by programmers to perform common tasks without having to write the code from scratch. There are few examples of library functions -

1. printf used to print formatted output to the console .

2. scanf used to read formatted input from

Teacher's Signature.....

Date 8/1/11
the console.

3. strlen: used to calculate the length of a string.

4. sqrt: used to calculate the square root of a number.

Answer No - 26

The output of the given programme will be:
C is placement oriented language it is C++.

Answer No - 27

The statement printf("./.d ./d"); and a and b) is a bit unusual and may not produce the expected output.

Answer No - 28

The output of the given programme will be:
"C ./ Far ./ Placement."

Answer - 29

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

```
int main() {
```

```
    float distance, time, speed;
```

```
    printf("Enter the distance between IITA University and Delhi in kilometers.");
```

```
    scanf("%f", &distance);
```

```
    time = 4; // Assuming the time taken to reach
```

Teacher's Signature.....

Delhi by bus is 4 hours
speed = distance / time

Print ("The speed of the bus is %.2f Km/h\n")
speed

return 0;

3

Answer-30

#include < stdio.h >

int main () {

int marks - satyam = 50;

int marks - suman = 70;

int marks - shyam = 80;

float average - marks ;

average - marks = (marks - satyam + marks - suman +
marks - shyam) / 30

printf ("The average marks of satyam , suman and shyam
is %.2f /n" , average - marks);

return 0;

3

Answer-31

#include < stdio.h >

int main () {

float speed = 4.0 // Speed in Km/h

float time = 3.0 / 60.0; // time in hours

float distance

distance = Speed * time;

printf ("The distance traveled is %.2f Km\n", distance)

return 0;

3

ANSWER - 31

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

```
int main()
```

```
int money_saurav, money_sejal, temp;
```

```
printf("Enter the amount of money given to  
Saurav:");
```

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

```
printf("Enter the amount of money given to Sejal:");
```

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

```
// swapping the values
```

```
temp = money_saurav;
```

~~money_saurav = money_sejal;~~~~money_sejal = temp;~~

```
printf("After rectifying the mistake, the amount  
of money given to Saurav is %d\n", money_saurav);
```

```
printf("After rectifying the mistake, the amount of  
money given to Sejal is %d\n", money_sejal);
```

Teacher's Signature.....

Answer no. 33

Yes, multiple escape sequences can be combined in a single line of program code. For example, you can use "\n\t" to represent a new line followed by a tab.

Answer no. 34

Comments in C programming are lines of code that are not executed by the compiler. They are used to provide explanations or notes about the code to make it more readable and understandable.

In C, there are two types of comments:

- 1. Single-line comments: These comments begin with "//" and continue until the end of the line. For example:

/*c

// This is a single-line comment.

- multi-line comments: These comments start with "/*" and end with "*/". They can span multiple lines. For example:

/*e

/* This is a
multiple-line comment */

To insert a comment in a C program, simply add the appropriate comment syntax at the desired location in the code.

Answer no. 35

The output of the given code will be "Yes". This is because the size of (int) will always be a positive value, so the condition size of (int)>-1 will evaluate to true.

Answer - 35

The issue in the statement is that the variable "number" should be passed as the address to the scanf() function using the "&" operator.

The correct statement would be: scanf ("%d", &number)

Answer - 37

The variable name, "gross-salary" is invalid because it contains a hyphen, which is not allowed in variable name in C programming. The variable name arg is also invalid because it contains a period, which is not allowed in variable names. The variable names "INTEREST", salary of emp", and "thereisbookinmybou" are all valid.

Answer - 38

#include <stdio.h>

Teacher's Signature.....

```
int main() {
```

 float tank_size = 175.0 / / gallons

 float drain_rate = 25.0 / / gallon per hour

 float time_required = tank_size / drain
 Rate / / hours

 Print f ("The time required to completely
 clean the tank is y. 2f hours / h", time
 required)

 return 0;

Answer - 39

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

```
int main() {
```

 float percent = 75.0

 float battery_power,

 float hour = 0

 while (battery_power < percent) {

 battery_power = 0.2 * hours + 1 hours ++

 Print f ("The battery power will reach 75.0 / /
 after y. 1 hours / h", hours);

 return 0

Answer - 40

b. Interpreter

Answer - 41

/0

Answer - 42

% e

Teacher's Signature.....

Answer - 43

The data type that is not considered a basic data type in C is b. array. Arrays are a compound data type that can hold multiple values of a certain type.

Answer - 43

Array

Answer - 44

The output of the given code will be c. "hell" "g".

Answer - 45

Magnitude, 5

Answer - 46

Basic pay

Answer - 47

L

Answer - \$853

The output of the program will be -32766 since the integer data type is 2 bytes, the range of values it can hold is -32768 to 32767. The value 32770 exceeds this range, so it wraps around to the minimum value, which is -32768 and then increments by 1 to become -32766

Teacher's Signature.....

Answer-54

The output of the program will be "Temperature in Fahrenheit is 37.00". The expression $(9/5)*c + 32$ is evaluated using integer division, which means that the result of $(9/5)$ will be 1. multiplying 1 by c and adding 32 gives us 37.00