**Assignment**

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.

Ans :- #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. 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.

Ans :- #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;

}

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.

Ans :- #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 returned to Mr. X is: Rs %.2f\n", amount\_returned);

return 0;

}

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

Ans:- #include <stdio.h>

int main() {

printf("Name : Your Name\n");

printf("DOB : Your Date of Birth\n");

printf("Mobile : Your 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.

Ans:- #include <stdio.h>

int main() {

int integer;

char character;

float floating\_point;

printf("Enter an integer: ");

scanf("%d", &integer);

printf("Enter a character: ");

scanf(" %c", &character);

printf("Enter a floating point number: ");

scanf("%f", &floating\_point);

printf("\nInteger: %d\n", integer);

printf("Character: %c\n", character);

printf("Floating point number: %f\n", floating\_point);

return 0;

}

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

The sales total is : $ 172.53

Ans :- #include <stdio.h>

int main() {

float cost = 172.53;

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

return 0;

}

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.

Ans:- #include <stdio.h>

int main() {

float apples\_per\_person = 6.5;

int number\_of\_people = 3;

float total\_apples;

total\_apples = apples\_per\_person \* number\_of\_people;

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

return 0;

}

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

Ans:- #include <stdio.h>

int main() {

float value = 123.456;

printf("The value in exponential format is: %.2e\n", value);

return 0;

}

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

Ans:- #include <stdio.h>

int main() {

char mobile\_number[11];

printf("Enter your mobile number: ");

scanf("%s", mobile\_number);

printf("Your mobile number is: %s\n", mobile\_number);

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)

Ans:- #include <stdio.h>

int main() {

float population = 30000;

float first\_year\_increase = 0.2;

float second\_year\_increase = 0.3;

population = population + (population \* first\_year\_increase);

population = population + (population \* second\_year\_increase);

printf("The population of the city after two years is: %.0f\n", population);

return 0;

}

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

Ans:- #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. 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.

Ans:- #include <stdio.h>

int main() {

float basic\_pay, hra, ta, salary;

printf("Enter basic pay: ");

scanf("%f", &basic\_pay);

hra = 0.15 \* basic\_pay;

ta = 0.20 \* basic\_pay;

salary = basic\_pay + hra + ta;

printf("Salary: %.2f", 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.

Ans:- #include <stdio.h>

#include <math.h>

int main() {

float xp, yp, xq, yq, slope, angle;

printf("Enter coordinates of point P: ");

scanf("%f %f", &xp, &yp);

printf("Enter coordinates of point Q: ");

scanf("%f %f", &xq, &yq);

slope = (yq - yp) / (xq - xp);

angle = atan(slope) \* 180 / 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:

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.

Ans:- #include <stdio.h>

int main() {

float g1, g2, g3, g4, g5, c1, c2, c3, c4, c5, spi;

printf("Enter grade points and credits for 5 courses:\n");

scanf("%f %f %f %f %f %f %f %f %f %f", &g1, &c1, &g2, &c2, &g3, &c3, &g4, &c4, &g5, &c5);

spi = (g1 \* c1 + g2 \* c2 + g3 \* c3 + g4 \* c4 + g5 \* c5) / (c1 + c2 + c3 + c4 + c5);

printf("SPI: %.2f", spi);

return 0;

}

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

Ans:- #include <stdio.h>

int main() {

float c, l, f;

printf("Enter the speed of the wave: ");

scanf("%f", &c);

printf("Enter the wavelength of the wave: ");

scanf("%f", &l);

f = c / l;

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

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]

Ans:- #include <stdio.h>

#include <math.h>

int main() {

float u = 30, a = 5, s = 70, v;

v = sqrt(u \* u + 2 \* a \* s);

printf("Final velocity: %.2f m/s", v);

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 ]

Ans:- #include <stdio.h>

#include <math.h>

int main() {

float u = 0, a = 4, t = 3, v, s;

v = u + a \* t;

s = u \* t + 0.5 \* a \* pow(t, 2);

printf("Final velocity: %.2f m/s\n", v);

printf("Distance traveled: %.2f m", s);

return 0;

}

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

Ans:- #include <stdio.h>

int main() {

int roll\_number, sum;

printf("Enter your university roll number: ");

scanf("%d", &roll\_number);

sum = (roll\_number % 10000) / 1000 + (roll\_number % 1000) / 100 + (roll\_number % 100) / 10 + roll\_number % 10;

printf("Sum of the last four digits of your university roll number: %d", 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**

**Ans:-** **#include <stdio.h>**

**int main() {**

**float height\_cm = 170, weight\_kg = 70, height\_ft, weight\_lb;**

**height\_ft = height\_cm \* 0.0328;**

**weight\_lb = weight\_kg \* 2.20462;**

**printf("Height: %.2f feet\n", height\_ft);**

**printf("Weight: %.2f pounds", weight\_lb);**

**return 0;**

**}**

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

1. A character variable named option.
2. An integer variable sum initialized to 0
3. A floating point variable, product, initialized to 1

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

Ans:- #include <stdio.h>

int main() {

int numbers[9];

printf("Enter nine integers: ");

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

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

}

printf("Numbers: ");

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

printf("%d", numbers[i]);

if ((i + 1) % 3 == 0 && i != 8) {

printf(", ");

}

}

return 0;

}

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

Ans:- In C programming, **header files** are additional files that contain definitions of different functions and their associated variables

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);

}

Ans:- the output of the program will be 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);

}

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

ANS:- In C programming, **library functions** are pre-defined functions that are grouped together and placed in a common location called a library. [These functions perform specific operations and can be used to get predefined output instead of writing your own code to get those outputs](https://www.tutorialspoint.com/what-are-the-c-library-functions).

Here are four examples of library functions in C:

1. printf(): This function is used to print formatted output to the console.
2. scanf(): This function is used to read formatted input from the console.
3. strlen(): This function is used to find the length of a string.
4. pow(): This function is used to calculate the power of a number

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);

}

ANS:- The program will output 56 70 38.

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

ANS:- The statement printf("%d", scanf("%d%d", &a, &b)); is a single line of code that reads two integers from the user and prints the number of integers read to the console.

Q28. What will be the output of following program?

#include <stdio.h>

void main()

{

printf(" \"C %% FOR %% PLACEMENT\"");

}

ANS :- The output of the program will be "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.

Answer:

#include <stdio.h>

int main() {

float distance;

float time = 4.0;

float speed;

printf("Enter the distance between GLA University and Delhi in kilometers: ");

scanf("%f", &distance);

speed = distance / time;

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.

Answer: Certainly! Here’s a ‘C’ program that calculates the average marks of Satyam, Suman, and Shyam:

#include <stdio.h>

int main() {

int satyam\_marks = 50;

int suman\_marks = 70;

int shyam\_marks = 80;

float average;

average = (satyam\_marks + suman\_marks + shyam\_marks) / 3.0;

printf("The average marks of Satyam, Suman, and Shyam is %.2f.\n", average);

return 0;

}

Copy

In this program, we first declare three variables satyam\_marks, suman\_marks, and shyam\_marks to store the marks obtained by Satyam, Suman, and Shyam, respectively. We also declare a variable average to store the calculated average.

We then calculate the average marks by adding the marks of the three participants and dividing the sum by 3.0 to get a floating-point result.

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.

Answer: To rectify the mistake made by Mohan, we can use a temporary variable to swap the values of the two variables. Here’s a C program that does this:

#include <stdio.h>

int main() {

int saurav\_money = 100;

int sajal\_money = 200;

int temp;

printf("Before swapping:\n");

printf("Saurav has %d rupees.\n", saurav\_money);

printf("Sajal has %d rupees.\n", sajal\_money);

// Swap the values of the two variables

temp = saurav\_money;

saurav\_money = sajal\_money;

sajal\_money = temp;

printf("\nAfter swapping:\n");

printf("Saurav has %d rupees.\n", saurav\_money);

printf("Sajal has %d rupees.\n", sajal\_money);

return 0;

}

Copy

In this program, we first declare two variables saurav\_money and sajal\_money to store the amount of money given to Saurav and Sajal, respectively. We also declare a temporary variable temp.

We then print out the initial values of the two variables using printf().

To swap the values of the two variables, we first assign the value of saurav\_money to temp. We then assign the value of sajal\_money to saurav\_money, and finally assign the value of temp to sajal\_money. This effectively swaps the values of the two variables.

Finally, we print out the new values of the two variables using printf().

I hope this helps Mohan rectify his mistake!

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.

Answer:

To calculate the distance travelled by you, we can use the formula:

distance = speed x time

Copy

where speed is your running speed in km/h and time is the time taken to reach the mess in hours.

Given that you ran at a speed of 4 km/h and it took you 3 minutes (0.05 hours) to reach the mess, we can calculate the distance travelled as:

distance = 4 x 0.05

Copy

Therefore, the distance travelled by you is **0.2 km**.

Here’s a C program that calculates the distance travelled:

#include <stdio.h>

int main() {

float speed = 4; // km/h

float time = 0.05; // hours

float distance = speed \* time;

printf("The distance travelled by you is %.2f km.", distance);

return 0;

}

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

Answer: Yes, it is possible to combine two or more escape sequences such as \n and \t in a single line of program code in C. For example, the following code will print “Hello” on one line, followed by a tab character, and then “World” on the next line:

printf("Hello\tWorld\n");

Copy

In this example, the \t escape sequence inserts a tab character between “Hello” and “World”, and the \n escape sequence moves the cursor to the next line.

Therefore, it is possible to combine multiple escape sequences in a single line of program code in C.

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

Answer: In C programming, **comments** are used to add notes or explanations to the code. They are ignored by the compiler and do not affect the execution of the program.

There are two types of comments in C:

* **Single-line comments**: These comments start with // and continue until the end of the line. They are used to add a short note or explanation to a single line of code.
* **Multi-line comments**: These comments start with /\* and end with \*/. They can span multiple lines and are used to add longer notes or explanations to a block of code.

Here are some examples of how to use comments in C:

// This is a single-line comment

int main() {

/\* This is a multi-line comment

that spans multiple lines \*/

return 0;

}

Copy

To insert a comment in a C program, simply add // or /\* ... \*/ before the text you want to comment.

B

Q35. What is wrong in this statement? scanf(“%d”,number);  
Answer: The statement scanf(“%d”,number); is incorrect because the scanf() function requires the address of the variable where the input value will be stored, not the variable itself.

To fix this error, we need to pass the address of the variable number to scanf() using the & operator. Here’s the corrected statement:

scanf("%d", &number);

Copy

Therefore, the correct statement should be scanf("%d", &number);.

Q36. What will be the output?

#include <stdio.h>

int main()

{

if (sizeof(int) > -1)

printf("Yes");

else

printf("No");

return 0;

}

Answer: The output of the program will be **Yes**.

In C, the sizeof operator returns the size of a data type in bytes. [The result of sizeof is an unsigned integer of type size\_t, which is always non-negative](https://stackoverflow.com/questions/34151309/why-is-sizeofint-1-false)[1](https://stackoverflow.com/questions/34151309/why-is-sizeofint-1-false).

In the given program, the expression sizeof(int) > -1 is equivalent to sizeof(int) > (unsigned int)-1. Since -1 is converted to an unsigned integer before the comparison, it becomes a very large number that is greater than the size of an integer on most implementations. Therefore, the condition in the if statement is true, and the program prints “Yes” to the console.

Therefore, the output of the program will be **Yes**.

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

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

Answer: Out of the given variable names, **gross-salary** and **thereisbookinmysoup** are invalid because they contain hyphens and spaces, respectively. Variable names in C must not contain hyphens or spaces. Instead, we can use underscores to separate words in a variable name.

The other two variable names, **salary of emp** and **avg.**, are also invalid. Variable names in C must not contain spaces or periods. Instead, we can use underscores to separate words in a variable name.

Therefore, the correct answer is: **gross-salary** and **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.

Answer:

#include <stdio.h>

int main() {

float total\_amount\_of\_water = 175;

float rate\_of\_drainage = 25;

float time = total\_amount\_of\_water / rate\_of\_drainage;

printf("The time required to completely clean the tank is approximately %.2f hours.", time);

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%?

Answer:

#include <stdio.h>

int main() {

float x = (1 - 0.75) / (-0.2);

printf("The battery power will be at 75%% after %.2f hours.", x);

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

Answer: [The tool used to convert high-level language into machine language in a single go is a **compiler** 1](https://www.bbc.co.uk/bitesize/guides/zmthsrd/revision/1)[2](https://www.geeksforgeeks.org/language-processors-assembler-compiler-and-interpreter/)[3](https://www.thegeeksclan.com/program-translator-tools/).

A **compiler** is a program that takes the entire source code of a program written in a high-level language and translates it into machine code all at once. [The resulting machine code can be executed directly on the target computer without any further processing 1](https://www.bbc.co.uk/bitesize/guides/zmthsrd/revision/1)[2](https://www.geeksforgeeks.org/language-processors-assembler-compiler-and-interpreter/)[3](https://www.thegeeksclan.com/program-translator-tools/).

On the other hand, an **interpreter** translates and executes the source code of a program line by line, without generating an executable file. [This means that the source code must be interpreted every time the program is run, which can make it slower than compiled code 4](https://stackoverflow.com/questions/19506164/what-is-the-process-of-translating-high-level-language-into-machine-language).

A **linker** is a program that combines object files generated by a compiler into a single executable file. [It resolves references between different object files and libraries, and generates an executable file that can be run on the target computer 5](https://edurev.in/question/1834941/Which-among-these-language-translators-that-can-convert-high-level-language-program-into-machine-lan).

An **assembler** is a program that translates assembly language into machine code. Assembly language is a low-level programming language that is easier to read and write than machine code, but still requires detailed knowledge of the underlying hardware architecture .

Therefore, the correct answer is **a. Compiler**.

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

a.%0 b.%d

c. %o d. %e

Answer: The format specifier for an octal number is **%o**.

[In C, the %o format specifier is used to print an integer in octal (base 8) format 1](https://www.geeksforgeeks.org/octal-numbers-c/)[2](https://www.includehelp.com/c/octal-literals.aspx). To print an integer in octal format, we can use the following syntax:

printf("%o", num);

Copy

where num is the integer to be printed in octal format.

Therefore, the correct answer is **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

Answer: The format specifier used to print the exponent value up to 2 decimal places is **d.%.2e**.

[The %e format specifier is used to print a floating-point number in scientific notation with an exponent 1](https://www.geeksforgeeks.org/format-specifiers-in-c/). [To print the exponent value up to 2 decimal places, we can use the precision specifier .2 along with %e, like this: %.2e 2](https://stackoverflow.com/questions/2538787/how-to-print-a-float-with-2-decimal-places-in-java).

Therefore, the correct answer is **d.%.2e**.

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

a. char

b. array

c. float

d. int

Answer: [A **data type** is a classification of data that tells the compiler or interpreter how the programmer intends to use the data 1](https://press.rebus.community/programmingfundamentals/chapter/data-types/).

Out of the given options, **array** is not a basic data type. [An **array** is a collection of elements of the same type, such as a collection of integers or characters 2](https://www.toppr.com/guides/computer-science/introduction-to-c-/data-types-variables-and-constants/concept-of-data-types/). It is a compound data type that can be built using basic data types.

On the other hand, **char**, **float**, and **int** are all examples of basic data types.

* [**char** represents individual characters 2](https://www.toppr.com/guides/computer-science/introduction-to-c-/data-types-variables-and-constants/concept-of-data-types/).
* [**float** represents fractional numbers 2](https://www.toppr.com/guides/computer-science/introduction-to-c-/data-types-variables-and-constants/concept-of-data-types/).
* [**int** represents whole numbers](https://press.rebus.community/programmingfundamentals/chapter/data-types/)[2](https://www.toppr.com/guides/computer-science/introduction-to-c-/data-types-variables-and-constants/concept-of-data-types/).

Therefore, the correct answer is **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

Answer: The output of the program will be **hello7**.

The first printf() statement in the program prints the string "hello" to the console and then moves the cursor back one position using the backspace escape sequence \b. This means that the letter o is overwritten by the double quote ", resulting in the output hell". However, since we are using double quotes to enclose the string, we need to escape them using a backslash \. Therefore, the correct string to print is "hell\"".

The printf() function returns the number of characters printed, which is 7 in this case. This value is then assigned to the variable x.

The second printf() statement in the program prints the value of x to the console, which is 7.

Therefore, the complete output of the program is **hello7**.

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

Answer:

#include <stdio.h>

int main() {

int b, c = 5;

printf("%d, %d", b, c);

return 0;

}

Q46. Which of the following is an identifier?

a. &fact b. Basic\_pay c. enum d. 1sum

Answer: An **identifier** is a name given to entities such as variables, functions, structures, etc. [that are created to give a unique name to an entity to identify it during the execution of the program 1](https://www.programiz.com/c-programming/c-keywords-identifier)[2](https://en.wikipedia.org/wiki/Identifier_%28computer_languages%29)[3](https://www.thoughtco.com/definition-of-identifier-958092)[4](https://www.techopedia.com/definition/1810/identifier-c).

Out of the given options, **Basic\_pay** and **enum** are identifiers because they follow the rules for naming identifiers. Identifiers can have letters (both uppercase and lowercase letters), digits, and underscores. The first letter of an identifier should be either a letter or an underscore. You cannot use keywords like int, while, etc. [as identifiers 1](https://www.programiz.com/c-programming/c-keywords-identifier)[2](https://en.wikipedia.org/wiki/Identifier_%28computer_languages%29)[3](https://www.thoughtco.com/definition-of-identifier-958092)[4](https://www.techopedia.com/definition/1810/identifier-c).

Therefore, the correct options are **b. Basic\_pay** and **c. enum**.

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

#include<stdio.h> char x, a=’c’;

x=printf("%c",a);

void main()

{

printf(“%d”,x);

}

a. c1 b. cgarbage

c. 1 c. c

answer: The output of the program will be **c1**.

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

1. (365.55)10 = (?)2
2. (453.65)10 = (?)8
3. (5164.12)10 = (?)16
4. (23.65)10 = (?)5
5. (772)10 = (?)7

Q49. Covert the following numbers to decimal number system-

1. (325.54)6 = (?)10
2. (1001010110101.1110101)2 = (?)10
3. (742.72)8 = (?)10
4. (AC94.C5)16 = (?)10

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

(DB56.CD4)16 = (?)2, (?)8, (?)4

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

(473.42)8 = (?)2, (?)10, (?)16, (?)5

Q52. Find the value of A?

1. (23)10 = (17)A
2. (21)16 = (41)A
3. (32)8 = (101)A

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

void main(){

int a=32770;

printf(“%d”,a);

}

Answer: The output of the program will be **-32766**.

The reason for this is that the integer a is assigned a value of 32770, which is greater than the maximum value that can be stored in a 2-byte integer. The maximum value that can be stored in a 2-byte integer is 32767. When an integer overflows, it wraps around to the minimum value that can be stored in a 2-byte integer, which is -32768. Therefore, the value of a becomes -32766 (-32768 + 2), and this is what gets printed by the printf() statement.

I hope this helps!

Q54: #include <stdio.h>

int main()

{

float c = 5.0;

printf ("Temperature in Fahrenheit is %.2f", (9/5)\*c + 32);

return 0;

}

Answer:

#include <stdio.h>

int main() {

float c = 5.0;

printf("Temperature in Fahrenheit is %.2f", (9.0/5.0)\*c + 32);

return 0;

}