



QUIZ

1. What will be the output of the following C application?

```
#include <stdio.h>

unsigned char fct(unsigned short int x)
{
    unsigned char b = 0, i, *px;
    px = (unsigned char*)&x;

    for (i = 0; i < sizeof(x); i++)
        if (px[i] & 0x01)
            b++;
    return b;
}

int main()
{
    unsigned short int n = 127;

    unsigned char result = fct(n);
    printf("result = %hhu\n", result);

    return 0;
}
```

- a. result = 3
- b. result = 1
- c. result = 4
- d. result = 127

2. What will be the output when the below C application will run?

```
#include <stdio.h>

int main()
{
    char a[] = { 'A', 'B', 'C' };
    char *pa = NULL, *p2a = NULL;

    pa = a + 1;
    p2a = a + 2;
    printf("*pa = %c, *p2a = %c\n", *pa, *p2a);

    return 0;
}
```

- a. *pa = B, *p2a = C
- b. *pa = B, *p2a = B
- c. *pa = C, *p2a = B
- d. *pa = C, *p2a = C



3. What will be the output of the below C application?

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

int main()
{
    char a[] = { 'A', 0x00, 'B', 0x00, 'C', 0x00};
    char* pa = NULL;

    pa = a;

    printf("array size = %u, string length = %u\n",
           (unsigned int)sizeof(a), (unsigned int)strlen(pa));

    return 0;
}
```

- a. array size = 3, string length = 3
- b. array size = 6, string length = 1
- c. array size = 1, string length = 3
- d. array size = 2, string length = 1

4. What will be the output of the below C application?

```
#include <stdio.h>

int main()
{
    char a[] = { 'A', 'B', 'C', 0, '\0'};
    int* pa = NULL;

    pa = (int*)a;

    printf("%s\n", (char*)pa);

    return 0;
}
```

- a. ABC
- b. ABC0
- c. ABC00
- d. ABC\0

5. What will be the output when the below C application will run?

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



```
int main()
{
    char a[] = { '-', '1', '2', 0, '.', '3', '4', 0 };
    printf("%.2f\n", (float)atof(a + 4));
    return 0;
}
```

- a. -12.00
- b. -120.34
- c. 0.34
- d. -120.00

6. What will be the output when the below C application will run?

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

void f(int* x)
{
    x[0] = 10;
}

int main()
{
    int a = 65;
    int *pa = &a;

    pa = (int*)malloc(sizeof(int));

    f(pa);
    printf("*pa = %d\n", *pa);

    free(pa);

    return 0;
}
```

- a. *pa = 10
- b. *pa = *pa
- c. *pa = 0
- d. *pa = 65

7. What will be the output of the following C application?

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



```
void fct(char *s1, char *s2, int idx)
{
    if (idx >= 0)
    {
        s1[idx] = s2[idx];
        fct(s1, s2, idx - 1);
    }
}

int main()
{
    char string1[] = "Exam_1";
    char string2[] = "Exam_2";

    int min_length = (strlen(string1) < strlen(string2)) ?
        strlen(string1) : strlen(string2);

    fct(string1, string2, min_length - 1);

    printf("string1 = %s, string2 = %s\n", string1, string2);

    return 0;
}
```

- a. string1 = Exam_2, string2 = Exam_1
- b. string1 = Exam_2, string2 = Exam_2
- c. string1 = Exam_1, string2 = Exam_1
- d. string1 = Exa2_2, string2 = Exam_2

8. What will be the output when the below C application will run?

```
#include <stdio.h>

#define COMP(A, B) ((A) * 3 > B ? (A) : (B))

int main()
{
    int x = 3, y = 3, z = 10, t = 5;

    printf("result = %d\n", COMP(x + y, z + t));

    return 0;
}
```

- a. result = 18
- b. result = 15
- c. result = 6
- d. result = 10



9. What will be the output when the below C application will run?

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

int main()
{
    int x = 2, y = 1;
    int *px = NULL, *py = NULL;

    px = (int*)malloc(x * sizeof(int));
    px[0] = 3;
    px[1] = 5;
    py = px + 1;

    int m = x * (py - px) < y * *py ? *px : *py;

    printf("\'m = %d\'\n", m);

    free(px);

    return 0;
}
```

- a. m = 3
- b. 'm = 5'
- c. m = 5
- d. 'm = 3'

10. Consider the following content saved into a file named `Points.txt`:

```
1 -3,-2,4,3
2 3,4,-1,-3
3
```

What will be the output of the following C application?

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

#define BUFFER_SIZE 256

typedef struct point point;
typedef struct rect rect;

struct point {
    int x, y;
};
```



```
struct rect {
    point point1;
    point point2;
};

int main()
{
    FILE* pf = fopen("Points.txt", "r");
    rect r;
    char buffer[BUFFER_SIZE], *token, list[] = ",\n";
    long size = 0;

    fseek(pf, 0, SEEK_END);
    size = ftell(pf);

    if (size <= BUFFER_SIZE)
    {
        fseek(pf, 0, SEEK_SET);
        fread(buffer, sizeof(buffer), 1, pf);

        token = strtok(buffer, list);

        while (token)
        {
            r.point1.x = atoi(token);
            token = strtok(NULL, list);
            r.point1.y = atoi(token);
            token = strtok(NULL, list);
            r.point2.x = atoi(token);
            token = strtok(NULL, list);
            r.point2.y = atoi(token);

            printf("diff_x = %d, diff_y = %d\n",
                abs(r.point2.x - r.point1.x),
                abs(r.point2.y - r.point1.y));

            token = strtok(NULL, list);
        }
    }

    return 0;
}
```

- a.
diff_x = 7, diff_y = 5
diff_x = 4, diff_y = 4
- b.
diff_x = 7, diff_y = 7
diff_x = 4, diff_y = 4
- c.
diff_x = 5, diff_y = 5
diff_x = 4, diff_y = 7
- d.
diff_x = 7, diff_y = 5
diff_x = 4, diff_y = 7



11. What will be the output of the following C++ program?

```
#include <iostream>
using namespace std;

class A
{
public:
    virtual int f() { return 1; }
    int g() { return 2; }
};

class B : public A
{
public:
    int f() { return 3; }
    int g() { return 4; }
};

int main()
{
    A* x = new B();
    cout << x->f() << x->g();
    return 0;
}
```

- a. 14
- b. 34
- c. 12
- d. 32

12. Taking into consideration the following C++ code, what will be the message printed to the console?

```
#include <iostream>

class A
{
    int x;
    void f() {}
};

class B {
    int y;
    virtual void f() {}
};

int main()
{
    A a; B b;
    std::cout << (sizeof(b) <= sizeof(a) ? "Yes" : "No");
    return 0;
}
```



- a. 0
- b. Yes
- c. YesNo
- d. No

13. What will be the output of the next C++ program?

```
#include <iostream>
using namespace std;

class X { public: virtual int function() = 0; };
class Y : public X
{
public:
    int function() { return 1; }
};
class Z : public Y
{
public:
    int function() { return 2; }
};

int main()
{
    X* x; Y y; Z z;
    y.function() ? x = &z : x = &y;
    cout << x-> function ();
    return 0;
}
```

- a. 1
- b. compile/runtime error
- c. 2
- d. 0

14. What will the following C++ code print?

```
#include <iostream>
using namespace std;

class Foo
{
public:
    Foo() { cout << "A"; }
    ~Foo() { cout << "B"; }
};

int main()
{
    Foo f1, f2;
    Foo* pf = new Foo();
    return 0;
}
```




- a. it will generate a runtime error
- b. AAABBB
- c. AAABB
- d. AABB

15. What will be the output of the following C++ program?

```
#include <iostream>
using namespace std;

class Student
{
public:
    bool graduated = true;
};

int main()
{
    Student s;
    Student* ps = &s;
    cout << (*ps).graduated;
    return 0;
}
```

- a. runtime error
- b. 0
- c. compile error
- d. 1

16. Given the C++ class S from below and assuming that we have included the necessary classes and libraries, what will be the output of the main function?

```
class S
{
public:
    string name;
    char operator[](int i) {
        return name[i];
    }
};

int main()
{
    S s;
    s.name = "ABC";
    s[0] = 'C';
    cout << s[0];
    return 0;
}
```



- a. runtime error
- b. C
- c. compile error
- d. A

17. What will be the output of the following C++ code sequence?

```
#include <iostream>
using namespace std;

class Foo { public: bool x = false; };

int main()
{
    Foo** v = new Foo * [1];
    v[0] = new Foo();
    cout << (*v)->x;
    return 0;
}
```

- a. the address of the x attribute
- b. 1
- c. runtime error
- d. 0

18. Assuming that B is a valid C++ class, what operator(s) could be called in the second line of code from the main function?

```
int main()
{
    B b;
    b * 2;
    return 0;
}
```

- a. multiplication operator or implicit cast
- b. multiplication operator or explicit cast
- c. only the multiplication operator
- d. only the copy constructor

19. What is the main reason why the overloading of the << operator for printing information to the console needs to be done using a friend function and not a method?

- a. because this is a unary operator
- b. in order for the operator to accept chained calls
- c. because the first operand is not the same type as the class
- d. in order to get access to the private attributes of the class



20. Giving the C++ classes from below and assuming that we have included the necessary classes and libraries, what will be the output when the main function is executed?

```
class A
{
public:
    ~A() { cout << "A"; }
};
class B
{
public:
    ~B() { cout << "B"; }
};

class C : B, A
{
public:
    ~C() {}
};

int main()
{
    C c;
    return 0;
}
```

- a. there will be no output
- b. a compile error because the inheritance is private
- c. AB
- d. BA

21. Which of the following functions can be used with an assignment operator in a PL/SQL statement?

- a. UPPER
- b. SUM
- c. COUNT
- d. MIN

22. Which of the following statements about implicit cursors are true?

- a. Are automatically opened and closed
- b. Store information regarding processing DDL statements
- c. Have user defined names
- d. Can be processed using FOR



23. The employees table exists has the following columns: employee_id, first_name, last_name and hire_date and 50 rows. Given the Oracle SQL statement:

```
select 'ab' column1 from employees where 'a'='b';
```

Which of the following statements is true:

- a. The statement runs successfully and displays some values
- b. The statement does not run as there is no column 'ab'
- c. The statement runs successfully and displays no rows
- d. The statement does not run as there is no column 'a' or 'b'

24. What is the main purpose of the DECODE function in Oracle SQL?

- a. To convert a value from one data type to another
- b. To perform conditional processing
- c. To sort the result set in either ascending or descending order
- d. To calculate the aggregate of a specific set of values

25. What does a CASE statement return in SQL Oracle when none of the conditions are met, and there is no ELSE clause?

Example:

```
SELECT CASE WHEN salary <= 5000 THEN 'Low' WHEN salary <= 7000 THEN 'Medium'  
END FROM employees;
```

- a. NULL
- b. 0
- c. An error message
- d. FALSE

26. Given the query:

```
SELECT cust_last_name  
FROM Customers join Orders on Customers.Customer_ID = Orders.Customer_ID  
group by cust_last_name  
ORDER BY count(Order_date) DESC;
```

What will happen if you try to run the query?

Consider the tables Customers (columns: Customer_ID number, cust_last_name varchar2(100)) and Orders (columns: Order_ID number, Customer_ID number, Order_date date not null).



- a. The query will run successfully and return the last name of the customer with the highest Customer_ID.
- b. The query will run successfully and return the last names of customers in descending order of their total orders number.
- c. The query will fail.
- d. The query will run successfully and return the last name of the customer who has placed the most recent order.

27. Can a procedure in PL/SQL call a function?

- a. Yes, always
- b. No, never
- c. Only if the function doesn't modify the database
- d. Only if the procedure and function access the same tables

28. Can an Oracle SQL select statement directly call a PL/SQL function?

- a. Yes, if certain conditions are met
- b. No, never
- c. Only if the function doesn't alter the tablespace
- d. Only if the function is in another schema

29. The ALTER TABLE statement in SQL can be used to:

- a. Drop a constraint from an existing table.
- b. Insert data into the table.
- c. Create a new table.
- d. Update a row from the table.

30. What is the main purpose of a FOREIGN KEY in Oracle SQL?

- a. To ensure data consistency and integrity across tables.
- b. To speed up queries on a database.
- c. To increase the storage space in a database.
- d. To make a column's values unique.

*** Observation: Each question has same number of points.**



Question	Correct Answer
1	b
2	a
3	b
4	a
5	c
6	a
7	b
8	c
9	d
10	d
11	d
12	d
13	c
14	c
15	d
16	c
17	d
18	a
19	c
20	c
21	a
22	a
23	c
24	b
25	a
26	b
27	a
28	a
29	a
30	a