QUIZ

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

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

void fct(char *s1, char *s2, int idx)
{
        int min_length = (strlen(s1) < strlen(s2)) ? strlen(s1) : strlen(s2);
        if (idx < (int)strlen(s1) &&
            idx < (int)strlen(s2))
        {
                char temp = s1[idx];
                s1[idx] = s2[(min_length - 1) - idx];
                s2[(min_length - 1) - idx] = temp;

                fct(s1, s2, idx + 1);
        }
}

int main()
{
        char string1[] = "ABCD";
        char string2[] = "ABC";

        fct(string1, string2, 0);

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

        return 0;
}
```

a. string1 = DCBA, string2 = CBA
b. string1 = ABCD, string2 = CBA
c. string1 = CBAD, string2 = CBA
d. string1 = DCBA, string2 = ABC

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

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

int main()
{
        char a[] = { 'A', 'B', 0, 'C'};
        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 = 2
b. array size = 4, string length = 4
c. array size = 4, string length = 2
d. array size = 2, string length = 2

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

```
#include <stdio.h>

int main()
{
        int x = 2, y = 3;
        int *px = NULL, *py = NULL;

        px = &x;
        py = &y;
        *px += 3;
        *py += 5;

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

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

        return 0;
}
```

a. 'm = 8'
b. m = 5
c. m = 8
d. 'm = 5'

4. 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 = 6
b. result = 18
c. result = 15
d. result = 10

5. 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;

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

int main()
{
    unsigned short int n = 21;

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

    return 0;
}
```

a. result = 3
b. result = 0
c. result = 4
d. result = -1

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

```
#include <stdio.h>

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

        pa = (int*)a;

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

        return 0;
}
```

a. A
b. A0BC0
c. ABC
d. BC

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

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

void f(int** x)
{
        int t = **x;
        *x = (int*)malloc(sizeof(int));
        if (*x)
        {
                **x = t + 10;
        }
}

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

        f(&pa);

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

        return 0;
}
```

a. *pa = 0
b. *pa = 75
c. *pa = 10
d. *pa = 65

8. 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 + 2;
        p2a = pa - 1;
        printf("*pa = %c, *p2a = %c\n", *pa, *p2a);

        return 0;
}
```

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

9. 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>

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[32];

        while (fgets(buffer, sizeof(buffer), pf))
        {
                sscanf(buffer, "%d,%d,%d,%d\n",
                        &r.point1.x, &r.point1.y,
                        &r.point2.x, &r.point2.y);

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

        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

10. 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', '\0' };

        printf("%d\n", atoi(a));

        return 0;
}
```

a. -12
b. -123
c. -12030
d. -123.0

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

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

class Car
{
public:
        string producer = "Dacia";
};
int main()
{
        Car c;
        Car* pc = &c;
        cout << pc->producer;
        return 0;
}
```

a. runtime error
b. Dacia
c. an empty string
d. compile error

12. What is the main reason why the overloading of the >> operator for reading information from the console needs to be done using a friend function and not a method?

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

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

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

class Foo {
public:
    Foo() { }

    Foo(const Foo& f) {
      cout << "#";
    }

    Foo operator=(Foo f) {
      cout << "@";
      return *this;
    }
};

int main() {
    Foo f, f3;
    Foo f2 = f;
    f3 = f2;
}
```

a. #@
b. ##@#
c. ##@
d. @@

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

```
#include <iostream>
using namespace std;
class Foo { public: int x = 123; };

int main()
{
        Foo f;
        Foo* pf = &f;
        f.x = 456;
        cout << pf->x;
}
```

a. the address of f
b. 456
c. 123
d. the address of x

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

```
#include <iostream>

class One
{
        int x, y; void f() {}
};
class Two {
        int a, b; virtual void f() {}
};

int main()
{
        std::cout << (sizeof(Two) > sizeof(One) ? "Yes" : "No");
        return 0;
}
```

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

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

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

class Base {
public:
    virtual void show() {
```

```
        cout << "Base" << endl;
    }
};

class Derived : public Base {
public:
    void show() {
        cout << "Derived" << endl;
    }
};

int main() {
    Base *x = new Derived();
    x->show();
    delete x; return 0;
}
```

a. Derived
b. Base
c. BaseDerived
d. it will generate a runtime error

17. Giving 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[1];
        return 0;
}
```

a. B
b. compile error
c. A
d. C

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. only the subtraction operator
b. only the copy constructor
c. subtraction operator or explicit cast
d. subtraction operator or implicit cast

19. 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 Base {
public:
    Base() {
        cout << "A";
    }

    ~Base() {
        cout << "B";
    }
};

class Derived : public Base {
public:
    Derived() {
        cout << "X";
    }
    ~Derived() {
        cout << "Y";
    }
};

int main() {
    Base *b = new Derived();
    delete b;
    return 0;
}
```

a. AXYB
b. AXB
c. XAYB
d. ABXY

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

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

```
class Foo
{
public:
        virtual int f() { return 100; }
        int g() { return 500; }
};

class Boo : public Foo
{
public:
        int f() { return 200; }
        int g() { return 600; }
};

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

a. 100600
b. 200500
c. 100500
d. 200600

21. Can an Oracle SQL select statement directly call a PL/SQL function?
a. Only if the function doesn't alter the tablespace
b. No, never
c. Yes, if certain conditions are met
d. Only if the function is in another schema

22. What distinguishes a correlated subquery from a non-correlated subquery in SQL?
a. The execution frequency of the subquery depends on the total number of rows in the database table
b. A correlated subquery is executed once for each row processed by the outer query
c. A correlated subquery does not have access to data from the outer query
d. A correlated subquery can only be used with SELECT statements

23. Which of the following functions can be used with an assignment operator in a PL/SQL statement?
a. COUNT
b. SUM
c. NVL
d. MAX

24. Consider the tables Customers (columns: Customer_ID, cust_last_name) and Orders (columns: Order_ID, Customer_ID, Order_date).
Given the query:

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

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

25. The ALTER TABLE statement in SQL can be used to:
a. Create a new table
b. Add a constraint to an existing table
c. Insert data into the table
d. Update a row from the table

26. Consider the following PL/SQL block:

```
declare
cursor c is select * from employees;
begin
open c;
for r in c loop
null;
end loop;
close c;
end;
/
```

The employees table exists and has the following columns: employee_id, first_name, last_name and hire_date. Which of the following statements is true?
a. The block will raise an exception only if the employees table is empty
b. The block will not compile
c. The block will run successfully
d. The block will always raise an exception

27. What is the purpose of the PRIMARY KEY constraint in SQL Oracle?
a. It ensures that the values in a column or group of columns match a list of specific values
b. It ensures that all values in a column or group of columns are different, and it cannot be left empty
c. It ensures that all values in a column or group of columns are the same
d. It ensures that a column or group of columns has values, and it cannot be left empty

28. Consider the following two tables:

```
Customers
CustomerID | CustomerName | ContactNumber
1 | Doe | 555-1234
2 | Smith | 555-2345
3 | Johnson | 555-3456
4 | Davis | 555-4567

Orders
OrderID | CustomerID | Product
1 | 1 | Apples
2 | 2 | Bananas
3 | 3 | Grapes
4 | 2 | Oranges
```

Given these two tables, what will be the result of the following SQL query?

```
SELECT Customers.CustomerName, Orders.Product
FROM Customers
RIGHT JOIN Orders
ON Customers.CustomerID = Orders.CustomerID;
```

a. A list of all customer ids who have placed orders, along with the product for each order they have placed
b. A list of all customers, with each customer's name listed only once, along with the product for each order they have placed, and NULL for customers who have not placed any orders
c. A list of all customers, with each customer's name listed once, along with a list of all products, regardless of whether the customer has placed an order or not
d. A list of all customers, with each customer's name possibly listed more than once, along with the product for each order they have placed, but no entries for customers who have not placed any orders

29. Can a procedure in PL/SQL call a function?
a. Only if the procedure and function access the same tables
b. Yes
c. No, never
d. Only if the function doesn't modify the database

30. 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 50 from employees where 1=20;
```

Which of the following statements is true:

a. The statement does not run as there is no column 100
b. The statement does not run as there is no column 20
c. The statement runs successfully and displays no rows
d. The statement runs successfully and displays some values

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

[redacted]

Cod Grilă

1	2	3	4	5	6
○	●	○	○	○	○

Disciplină

- ○ AP1
- ○ BT1
- ○ BT2
- ○ CSIE 1
- ○ CSIE 2
- ○ CSIE 3
- ● CSIE 4
- ○ CSIE 5
- ○ CIG1
- ○ CIG2
- ○ DA1
- ○ ETA1
- ○ EAM1
- ○ FIN1
- ○ MAN1
- ○ MK1
- ○ MK2
- ○ REI1
- ○ REI2
- ○ REI5
- ○ MD

30 x 3p = 90p

Nr.	A	B	C	D
1	○	○	●	○
2	○	○	●	○
3	○	○	○	●
4	○	○	●	○
5	○	●	○	○
6	●	○	○	○
7	○	●	○	○
8	●	○	○	○
9	○	○	○	●
10	●	○	○	○
11	○	●	○	○
12	○	●	○	○
13	○	●	○	○
14	○	●	○	○
15	●	○	○	○
16	●	○	○	○
17	●	○	○	○
18	○	○	○	●
19	○	●	○	○
20	○	●	○	○
21	○	○	●	○
22	○	●	○	○
23	○	○	●	○
24	○	●	○	○
25	○	●	○	○
26	○	○	○	●
27	○	●	○	○
28	○	○	○	●
29	○	●	○	○
30	○	○	●	○
31	○	○	○	○
32	○	○	○	○
33	○	○	○	○
34	○	○	○	○
35	○	○	○	○