Quiz 10

5.

What will be the output of the program ?

```
#include<stdio.h>

int main()
{
    enum days {MON=-1, TUE, WED=6, THU, FRI, SAT};
    printf("%d, %d, %d, %d, %d, %d\n", MON, TUE, WED, THU, FRI, SAT);
    return 0;
}
```

A. -1, 0, 1, 2, 3, 4

B. -1, 2, 6, 3, 4, 5

C. -1, 0, 6, 2, 3, 4

D. -1, 0, 6, 7, 8, 9

6.

Point out the error in the program?

```
struct emp
{
    int ecode;
    struct emp e;
};
```

A. Error: in structure declaration

B. Linker Error

C. No Error

D. None of above

The Questions:

1.

Which of the following special symbol allowed in a variable name?

A. * (asterisk)

B. | (pipeline)

C. - (hyphen)

D. _ (underscore)

2.

What is the similarity between a structure, union and enumeration?

A. All of them let you define new values

B. All of them let you define new data types

C. All of them let you define new pointers

D. All of them let you define new structures

3.

What will be the output of the program ?

```
#include<stdio.h>

int main()
{
    union a
    {
        int i;
        char ch[2];
    };
    union a u;
    u.ch[0]=3;
    u.ch[1]=2;
    printf("%d, %d, %d\n", u.ch[0], u.ch[1], u.i);
    return 0;
}
```

A. 3, 2, 515

B. 515, 2, 3

C. 3, 2, 5

D. 515, 515, 4

4.

What will be the output of the program ?

```
#include<stdio.h>

int main()
{
    union var
    {
        int a, b;
    };
    union var v;
    v.a=10;
    v.b=20;
    printf("%d\n", v.a);
    return 0;
}
```

A. 10

B. 20

C. 30

D. 0

7.

What is the output of the program given below ?

```c
#include<stdio.h>
int main()
{
    enum status { pass, fail, atkt};
    enum status stud1, stud2, stud3;
    stud1 = pass;
    stud2 = atkt;
    stud3 = fail;
    printf("%d, %d, %d\n", stud1, stud2, stud3);
    return 0;
}
```

A. 0, 1, 2
B. 1, 2, 3
C. 0, 2, 1
D. 1, 3, 2

8.

What is the output of the program

```c
#include<stdio.h>
int main()
{
    struct emp
    {
        char name[20];
        int age;
        float sal;
    };
    struct emp e = {"Tiger"};
    printf("%d, %f\n", e.age, e.sal);
    return 0;
}
```

A. 0, 0.000000
B. Garbage values
C. Error
D. None of above

9.

Assunming, integer is 2 byte, What will be the output of the program?

```c
#include<stdio.h>

int main()
{
    printf("%x\n", -1>>1);
    return 0;
}
```

A. ffff
B. fff
C. 0000
D. fff0

10.

If an `unsigned int` is 2 bytes wide then, What will be the output of the program ?

```c
#include<stdio.h>

int main()
{
    unsigned int m = 32;
    printf("%x\n", ~m);
    return 0;
}
```

A. ffff
B. 0000
C. ffdf
D. ddfd

11.

What will be the output of the program?

```c
#include<stdio.h>

int main()
{
    unsigned char i = 0x80;
    printf("%d\n", i<<1);
    return 0;
}
```

A. 0
B. 256
C. 100
D. 80

14.

What will be the output of the program?

```c
#include<stdio.h>
int main()
{
    int a = 500, b = 100, c;
    if(!a >= 400)
        b = 300;
    c = 200;
    printf("b = %d c = %d\n", b, c);
    return 0;
}
```

A. b = 300 c = 200

B. b = 100 c = garbage

C. b = 300 c = garbage

D. b = 100 c = 200

15.

What will be the output of the program?

```c
#include<stdio.h>
int main()
{
    unsigned int i = 65535; /* Assume 2 byte integer*/
    while(i++ != 0)
        printf("%d",++i);
    printf("\n");
    return 0;
}
```

A. Infinite loop

B. 0 1 2 ... 65535

C. 0 1 2 ... 32767 - 32766 -32765 -1 0

D. No output

16.

What will be the output of the program?

```c
#include<stdio.h>
int main()
{
    int x = 3;
    float y = 3.0;
    if(x == y)
        printf("x and y are equal");
    else
        printf("x and y are not equal");
    return 0;
}
```

A. x and y are equal

B. x and y are not equal

C. Unpredictable

D. No output

17.

Point out the error, if any, in the following for loop.

```c
#include<stdio.h>
int main()
{
    int i=1;
    for(;;)
    {
        printf("%d\n", i++);
        if(i>10)
            break;
    }
    return 0;
}
```

A. There should be a condition in the `for` loop

B. The two semicolons should be dropped

C. The `for` loop should be replaced with `while` loop.

D. No error

Quiz 10 solution

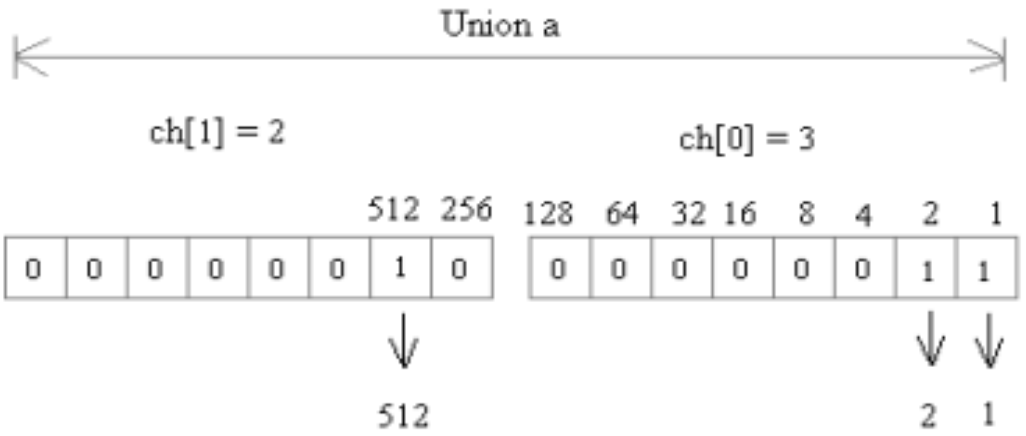
1. Option D
2. Option B
3.

Answer: Option **A**

Explanation:

The system will allocate 2 bytes for the union.

The statements `u.ch[0]=3; u.ch[1]=2;` store data in memory as given below.



So, 512 + 2 + 1 = 515

i = 515

4. Option B
5. Option D
6.

Answer: Option **A**

Explanation:

The structure `emp` contains a member `e` of the same type.(i.e) `struct emp`. At this stage compiler does not know the size of sttructure.

7.

Answer: Option **C**

Explanation:

`enum` takes the format like {0,1,2..) so `pass=0, fail=1, atkt=2`

`stud1 = pass` (value is 0)

`stud2 = atkt` (value is 2)

`stud3 = fail` (value is 1)

Hence it prints 0, 2, 1

8.

Answer: Option **A**

Explanation:

When an automatic structure is partially initialized remaining elements are initialized to 0(zero).

10. Option C

11. Option B

12. Option A

9.

Answer: Option **A**

Explanation:

Negative numbers are treated with 2's complement method.

1's complement: Inverting the bits (all 1s to 0s and all 0s to 1s)
2's complement: Adding 1 to the result of 1's complement.

```
Binary of 1(2byte)     :  0000 0000 0000 0001
Representing -1:
1s complement of 1(2byte)    : 1111 1111 1111 1110
Adding 1 to 1's comp. result : 1111 1111 1111 1111
Right shift 1bit(-1>>1): 1111 1111 1111 1111 (carry out 1)
Hexadecimal             : f   f    f    f
(Filled with 1s in the left side in the above step)
```

13. Option D

14. Option D

15. Option A

16. Option A

17.

Answer: Option **D**

Explanation:

Step 1: `for(;;)` this statement will generate infinite loop.
Step 2: `printf("%d\n", i++);` this statement will print the value of variable `i` and increement `i` by 1(one).
Step 3: `if(i>10)` here, if the variable `i` value is greater than 10, then the `for` loop breaks.

Hence the output of the program is

1
2
3
4
5
6
7
8
9
10