

# Set of multiple choice questions about programming with P17 prize

To become a true programmer, you first need to master the knowledge of this discipline. To help readers easily get acquainted with this subject, Network Administrator has synthesized and sent you extremely interesting multiple choice questions. Invite your reference.

1. Question 1. What does the following statement mean:

```
FILE * fopen (tepl, 'r +');
```

1. Open the allowable text file
  2. Open the readable text file
  3. Open the text file to allow both read and write
  4. Open an existing text file to read
2. Question 2. What does the following statement mean:

```
int open (tepl, O_TRUNC);
```

1. Open the binary file to write
  2. Delete the contents of the file
  3. Open the text file to read and write
  4. All the answers above
3. Question 3. The component's structure can be accessed through the pointer as (with p being the structure pointer, a is the structure component):
1. (\* p) .a;
  2. \* p-> a;
  3. 1 and 2 wrong
  4. 1 and 2 are correct
4. Question 4. Which statement is invalid in the following declaration:

```
float yi} t, * p, a [10];
```

1. p = & t;
  2. p = & t.x;
  3. p = & a [5];
  4. p = & a;
5. Question 5. Please choose the best sentence:

```
struct immediately {int ng, th, ni} vaotruong, ratruong; typedef struct s
```

1. No assignment: vaotruong = ratruong;
2. 'sinhvien' is the structure name, 'vaotruong', 'ratruong' is the structure variable name

3. You can write 'vaotruong.ng', 'ratruong.th', 'sinhvien.vaotruong.n' to access the corresponding components.

4. 1, 2, 3 right

6. Question 6. In value initializations for the following structures, select the correct sentence:

```
struct S1 {int immediately, ladder, male;}; S1 = (2, 1, 3);
```

```
struct S2 {char hoten [10]; struct S1 ngaysinh;}; struct S1 ngaysinh;}; S
```

1. S1 correct

2. S2 is correct

3. Both S1 and S2 are correct

4. Both S1 and S2 are wrong

7. Question 7. For structured types, which of the following is not allowed:

1. Assign variables to each other

2. Assign two array elements (structure type) to each other.

3. Assign an array element (structure type) to a variable or vice versa.

4. Assign two structured arrays with the same number of elements.

8. Question 8. Select the best sentence in the following sentences:

```
struct {int big; float soluong;} x[10]; float soluong; x [10];
```

```
for (int i = 0; i 10; i ++) scanf ('% d% f', & x [i] .to, & x [i] .soluong
```

1. The above program segment has a syntax error

2. Addressing operators are not allowed for 'to' and 'soluong' components

3. The address of the 'soluong' component leads to an improper operation

4. All is wrong

9. Question 9. To create a linked list, according to which of the following students is the correct declaration of the self-pointing structure to be used:

Student 1:

```
struct Sv * tiep;};
```

Student 2:

```
typedef
```

```
struct SV node;
```

```
node * tiep;};
```

Student 3:

```
typedef
```

```
struct struct * tiep;} node;
```

1. first

2. 2

3. 2 and 3

4. All 3 declare above
10. Question 10. To use the linked list, consider the following two statements (need 1KB to store information about a student):
- Declaration 1: `struct SV {thongtin;struct SV *tiep;}; struct SV * tiep;};`
- Declaration 2: `SV struct {thongtin}; struct DS *;;;`; (With 'thongtin' is a data component of the structure);

Choose the best sentence in the following sentences:

1. Declaration 1 takes more memory than declaring 2
2. Declaration 2 will help the program run faster when changing positions 2 students
3. Declaration 1 will help save the command more when writing the function to change the position of 2 students
4. Declaration 2 will help the program run faster when browsing the list

You finished reading the article "**Set of multiple choice questions about programming with P17 prize**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.