

NET2006: Object-Oriented Programming: Quiz 1

Given these classes, answer the following three questions.

```
class Date {
public:
    int nMonth, nDay, nYear;
};

class Person {
public:
    string sName, sAddress;
    Date dBirthDate;
};
```

1. Create an array of **Person** objects. The array will be called **Clients**, and it will contain 100 elements:

- a) `Person Clients[99];`
- b) `Person Clients[100];`
- c) `Person Clients[101];`
- d) `class Person Clients[99];`
- e) `Person.Clients[99];`

2. Assume **Clients** array has been initialized with data. To display the first client's name, we would execute the statement:

- a) `cout << Clients[1].Person.sName;`
- b) `cout << Clients.sName;`
- c) `cout << Clients[0].sName;`
- d) `cout << Clients.sName[0];`
- e) None of the above.

3. To put **9** into the day component of the third client's birthday, you would execute the statement:

- a) `Clients[2].Person.Date.dBirthDate.nDay = 9;`
- b) `Clients[3].dBirthDate.nDay = 9;`
- c) `Clients.dBirthDate.nDay[2] = 9;`
- d) `Clients[3].dBirthDate[nDay] = 9;`
- e) None of the above.

4. Given the following definitions, which one of the expressions is true:

```
int z = 1, y = 2, x = z+y;
```

- a) `0`
- b) `z < x || y < z`
- c) `z < x && y > x`
- d) `z > x && x > 0`
- e) `2 * x == y + 4 && y < z`

5. The line of code that declares / defines a character variable called **c** and an array of 100 characters called **acArray** is:

- a) `char c, acArray[]=100;`
- b) `char c, acArray[100];`
- c) `char c, 100[acArray];`
- d) `char c; acArray[100];`
- e) `char c, char acArray[100];`

6. You use **#include** to:

- a) Tell the compiler where program execution begins.
- b) Declare and define your variables.
- c) Add comments to your program.
- d) Tell the linker how to assemble all components of your project.
- e) None of the above.

7. Imagine you make a function call from **main()**, and this call passes a variable as an argument using pass-by-value:

- a) the function gets a copy of whatever was stored in the original variable in **main()**.
- b) the function returns the value of the argument (which may or may not have been modified).
- c) the function receives a variable which is really an *alias* to the original variable in **main()**.
- d) instructions in the function can change the value in the original variable in **main()** while executing.
- e) both c) and d).

8. Given the following, what will be displayed after these statements execute:

```
int nSpeed = 82;
if (nSpeed > 120)
    cout << "Fine: 150 ";
if (nSpeed < 105)
    cout << "Fine: 100 ";
if (nSpeed < 95)
    cout << "Fine: 75 ";
if (nSpeed < 75)
    cout << "Fine: 50 ";
else
    cout << "No Fine ";
```

- a) `Fine: 100`
- b) `Fine: 75`
- c) `Fine: 100 Fine: 75 No Fine`
- d) `Fine: 75 No Fine`
- e) `No Fine`

9. Which of the following is NOT a valid **int**?

- a) `0xdeaddeed`
- b) `111111111`
- c) `-858532192`
- d) `-2000000000`
- e) `0x002kez4a`

10. Which answer Identifies the correct size of each data type (in bytes):

- a) `int:2 float:4 long:4 char:1 double:8`
- b) `int:4 float:4 long:4 char:1 double:8`
- c) `int:4 float:6 long:4 char:1 double:8`
- d) `int:4 float:6 long:8 char:1 double:6`
- e) `int:2 float:4 long:4 char:2 double:12`

11. The three expressions (in their correct sequence) in a for statement are:

- a) test, increment, controller
- b) initialize, increment, test
- c) initialize, controller, test
- d) initialize, test, increment
- e) initialize, increment, controller

12. The expression **18%8** evaluates to:

- a) 10
- b) 8
- c) 2.25
- d) 2
- e) None of the above.