

1. Інкапсуляція дозволяє:

- ☐ Призначте значення змінним
- ☐ Оголосити функцію
- ☐ Сховати деталі реалізації класу

2. Ідея інкапсуляції полягає у:

- ☐ інкапсуляції атрибутів та методів об'єкта
- ☐ приховуванні все про об'єкт, виставляючи лише те, що необхідно
- ☐ створенні примірників класу
- ☐ наданні функцій користувачам класу

3. Введіть код, щоб оголосити клас MyClass, з одним членом даних з ім'ям myData типу string, а потім призначте йому значення "I love C++" з функції main. Не забудьте включити файл заголовка для даних типу string.

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

class MyClass
{
public:
    string myData;
};

int main()
{
    MyClass obj;
    obj.myData = "I love C++";
}
```

4. Заповніть пробіли, щоб оголосити клас MyClass, з одним приватним членом даних на ім'я myData типу int. Оголосити публічну функцію MyData (), який встановлює значення myData такому, що визначено його аргументом.

```
class MyClass
{
    ____:
    int myData;
    ____:
    void setMyData( ____ arg) {
        myData = arg;
    }
};
```

5. Заповніть пробіли, щоб оголосити метод `get` (отримання) для класу `MyClass` елементу `myData`. Метод `get` для `myData` - це публічна функція з назвою `getMyData ()`, яка повертає значення `myData`.

```
class MyClass
{
private:
    int myData;
public:
    ____ getMyData() {
        ____ myData;
    }
};
```

6. Введіть відсутні частини, щоб оголосити об'єкт класу `MyClass` та встановіть значення його елемента рівним 14, після чого вивести значення на екрані. Виконайте всі згадані операції через публічні функції `setMyData` та `getMyData`.

```
int main()
{
    MyClass obj;
    obj.setMyData( ____ );
    cout << ____ .getMyData() << endl;
}
```

7. Коли викликається конструктор?

- ☐ Коли створюється об'єкт класу
- ☐ Ніколи
- ☐ Коли клас оголошується

8. Заповніть пробіли, щоб оголосити клас `my_class` його конструктором.

```
_____ my_class
{
private:
int my_data;
public:
_____ (int arg) {
my_data = ____;
}
};
```

9. Заповніть пробіли, щоб оголосити клас `MyClass` з його конструктором. У функції `main` створіть об'єкт `MyClass` і передайте значення 14 як його аргумент конструктора

```
_____ MyClass {
private:
int myData;
public:
MyClass(int arg) {
myData = arg;
}
};
int main() {
MyClass obj( ____ );
}
```