**Câu 10:**

1/Class mô tả cấu trúc dữ liệu còn instance of class là 1 đối tượng của cấu trúc dữ liệu tồn tại trong bộ nhớ.

2/Với structure, ta không có phương thức để thực hiện các hành động như class mà chỉ có giá trị. Ngoài ra với class, giá trị khai báo luôn ở chế độ private nên chỉ các phương thức trong class đó mới truy xuất được đến các giá trị.

3/ private

4/ name: getRadius() class: Circle

5/ Sẽ giống thiết kế của blueprint.

6/ Accesor function: Không cần đối số, có cùng kiểu với biến trả về, tên bắt đầu bằng tiền tố Get và nó giúp cho các biến thành viên “tàng hình”, có thể lấy lại các biến private, không làm thay đổi giá trị dữ liệu của đối tượng.

Mutator function: Lưu giá trị trong biến thành viên private và có thể đổi giá trị.

7/ Nên để các biến thành viên trong private vì nó giúp hạn chế các truy xuất sửa đổi dữ liệu không mong muôn từ bên ngoài hay các hàm gọi bên ngoài.

8/ Sẽ tốt hơn nếu không khai báo cin và cout trong các statements của hàm vì điều đó nên được khai báo trong hàm main còn các hàm chức năng chỉ nên làm theo đúng nhiệm vụ của nó.

9/ Các biến dữ liệu vì đó là thông tin không nên thay đổi.

10/ Constructor là hàm khởi tạo cho class, destructor là hàm hủy class để tránh rò rỉ bộ nhớ.

11/ Default constructor là hàm khởi tạo mặc định giúp cho việc khởi tạo các giá trị mặc định cho class mà không phải do người dùng cung cấp. Chỉ có một hàm khởi tạo mặc định.

12/ Có thể có nhiều hơn 1 hàm constructor trong class nhưng chỉ có duy nhất một hàm destructor.

13/ Constructor vẫn thực thi nếu class là bộ nhớ động, chỉ cần khởi tạo pointers trong constructor, nó sẽ tự động phân bố trước không gian.

14/Khi gọi constructor mặc định.

15/ Là nghĩa vụ một class biết và có thể làm các nhiệm vụ.

16/ Định danh hết mọi danh từ và đại từ, sau đó lọc lại những class có liên quan đến problem.

**Câu 11:**

17/ functional promming/procedural programming, object oriented programming.

18/ functional promming/procedural programming.

19/ object oriented programming.

20/ encapsulation

21/ class.

22/ structure.

23/ access specifier

24/ private

25/ public

26/ instatiation

27/ ->

28/ Canine.h(header file)

29/ Canine.cpp

30/ inline

31/ constructor

32/ constructor

33/constructors

34/ return

35/ defaut

36/ destructor

37/ ~

38/ return

39/ defaut

40/ parameter

41/ contructor, destructor

42/ intialization

**Câu 12 :**

51/ F

52/ T

53/ F

54/ T

55/ F

56/ F

57/ T

58/ T

59/ F

60/ F

61/ T

62/ F

63/ T

64/ F

65/ T

66/ T

67/ F

68/ T

69/ T

70/ F

71/ F

72/ T

**Câu 13 :**

73/#include<iostream>

using namespace std ;

private :

public :

setCenter(double X, double Y)

setRadius(double radius)

thiếu ; khi kết thúc class

74/ sau khai báo class <Tên class> không có ;

private:

public:

thiếu ; khi kết thúc class

cout << và cin >>, chưa khai báo tên hàm lunar

75/ lỗi ở class tương tự câu 74, chỉ cần có 1 hàm setWeight là đủ, chưa thiết lập constructor trong class.

76/ Thiếu dấu ; khi kết thúc class. Trong void Change::Change thiếu kiểu truyền vào cho d và q.