**********OOP Basic Concepts"**********

1. Which was the first purely object oriented programming language developed?
a) Java
b) C++
c) SmallTalk
d) Kotlin
View Answer

Answer: c
Explanation: SmallTalk was the first programming language developed which was purely object oriented. It was dev eloped by Alan Kay. OOP concept came into the picture in 1970's.
2. Which of the following best defines a class?
a) Parent of an object
b) Instance of an object
c) Blueprint of an object
d) Scope of an object
View Answer

Answer: c
Explanation: A class is Blueprint of an object which describes/ shows all the functions and data that are provided by an object of a specific class. It can't be called as parent or instance of an object. Class in general describes all the pro perties of an object.
3. Who invented OOP?
a) Alan Kay
b) Andrea Ferro
c) Dennis Ritchie
d) Adele Goldberg
View Answer

Answer: a
Explanation: Alan Kay invented OOP, Andrea Ferro was a part of SmallTalk Development. Dennis invented C++ an d Adele Goldberg was in team to develop SmallTalk but Alan actually had got rewarded for OOP.

4. What is the additional feature in classes that was not in structures?
a) Data members
b) Member functions
c) Static data allowed
d) Public access specifier
View Answer

Answer: b
Explanation: Member functions are allowed inside a class but were not present in structure concept. Data members, s tatic data and public access specifiers were present in structures too.
5. Which is not feature of OOP in general definitions?
a) Code reusability
b) Modularity
c) Duplicate/Redundant data
d) Efficient Code
View Answer

Answer: c
Explanation: Duplicate/Redundant data is dependent on programmer and hence can't be guaranteed by OOP. Code r

eusability is done using inheritance. Modularity is supported by using different code files and classes. Codes are mor e efficient because of features of OOP.
Become Top Ranker in Object Oriented Programming Now!
6. Pure OOP can be implemented without using class in a program. (True or False)
a) True
b) False
View Answer

Answer: b
Explanation: It's false because for a program to be pure OO, everything must be written inside classes. If this rule is violated, the program can't be labelled as purely OO.
7. Which Feature of OOP illustrated the code reusability?
a) Polymorphism
b) Abstraction
c) Encapsulation
d) Inheritance
View Answer

Answer: d
Explanation: Using inheritance we can reuse the code already written and also can avoid creation of many new funct ions or variables, as that can be done one time and be reused, using classes.
8. Which language does not support all 4 types of inheritance?
a) C++
b) Java
c) Kotlin
d) Small Talk
View Answer

Answer: b
Explanation: Java doesn't support all 4 types of inheritance. It doesn't support multiple inheritance. But the multiple inheritance can be implemented using interfaces in Java.
9. How many classes can be defined in a single program?
a) Only 1
b) Only 100
c) Only 999
d) As many as you want
View Answer

Answer: d
Explanation: Any number of classes can be defined inside a program, provided that their names are different. In java , if public class is present then it must have the same name as that of file.
10. When OOP concept did first came into picture?
a) 1970's
b) 1980's
c) 1993
d) 1995
View Answer

Answer: a
Explanation: OOP first came into picture in 1970's by Alan and his team. Later it was used by some programming la nguages and got implemented successfully, SmallTalk was first language to use pure OOP and followed all rules stri ctly.
11. Why Java is Partially OOP language?
a) It supports usual declaration of primitive data types
b) It doesn't support all types of inheritance

c) It allows code to be written outside classes
d) It does not support pointers
View Answer

Answer: a
Explanation: As Java supports usual declaration of data variables, it is partial implementation of OOP. Because acco rding to rules of OOP, object constructors must be used, even for declaration of variables.
12. Which concept of OOP is false for C++?
a) Code can be written without using classes
b) Code must contain at least one class
c) A class must have member functions
d) At least one object should be declared in code
View Answer

Answer: b
Explanation: In C++, it's not necessary to use classes, and hence codes can be written without using OOP concept. C lasses may or may not contain member functions, so it's not a necessary condition in C++. And, an object can only b e declared in a code if its class is defined/included via header file.
13. Which header file is required in C++ to use OOP?
a) iostream.h
b) stdio.h
c) stdlib.h
d) OOP can be used without using any header file
View Answer

Answer: d
Explanation: We need not include any specific header file to use OOP concept in C++, only specific functions used i n code need their respective header files to be included or classes should be defined if needed.
14. Which of the two features match each other?
a) Inheritance and Encapsulation
b) Encapsulation and Polymorphism
c) Encapsulation and Abstraction
d) Abstraction and Polymorphism
View Answer

Answer: c
Explanation: Encapsulation and Abstraction are similar features. Encapsulation is actually binding all the properties i n a single class or we can say hiding all the features of object inside a class. And Abstraction is hiding unwanted dat a (for user) and showing only the data required by the user of program.
15. Which feature allows open recursion, among the following?
a) Use of this pointer
b) Use of pointers
c) Use of pass by value
d) Use of parameterized constructor
View Answer

Answer: a
Explanation: Use of this pointer allows an object to call data and methods of itself whenever needed. This helps us c all the members of an object recursively, and differentiate the variables of different scopes

**********"Classes"**********

1. Which of the following is not type of class?
a) Abstract Class

b) Final Class
c) Start Class
d) String Class
View Answer

Answer: c
Explanation: Only 9 types of classes are provided in general, namely, abstract, final, mutable, wrapper, anonymous, input-output, string, system, network. We may further divide the classes into parent class and subclass if inheritance is used.

2. Class is pass by _____
a) Value
b) Reference
c) Value or Reference, depending on program
d) Copy
View Answer

Answer: b
Explanation: Classes are pass by reference, and the structures are pass by copy. It doesn't depend on the program.

3. What is default access specifier for data members or member functions declared within a class without any specifier, in C++?
a) Private
b) Protected
c) Public
d) Depends on compiler
View Answer

Answer: a
Explanation: The data members and member functions are Private by default in C++ classes, if none of the access specifier is used. It is actually made to increase the privacy of data.
Note: Join free Sanfoundry classes at Telegram or Youtube
advertisement

4. Which is most appropriate comment on following class definition?

```
class Student
{
   int a;
   public : float a;
};
```

a) Error : same variable name can't be used twice
b) Error : Public must come first
c) Error : data types are different for same variable
d) It is correct
View Answer

Answer: a
Explanation: Same variable can't be defined twice in same scope. Even if the data types are different, variable name must be different. There is no rule like Public member should come first or last.

Take Object Oriented Programming Practice Tests - Chapterwise!
Start the Test Now: Chapter 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

5. Which is known as a generic class?
a) Abstract class
b) Final class
c) Template class
d) Efficient Code

View Answer

Answer: c
Explanation: Template classes are known to be generic classes because those can be used for any data type value and the same class can be used for all the variables of different data types.
6. Size of a class is _____
a) Sum of the size of all the variables declared inside the class
b) Sum of the size of all the variables along with inherited variables in the class
c) Size of the largest size of variable
d) Classes doesn't have any size
View Answer

Answer: d
Explanation: Classes doesn't have any size, actually the size of object of the class can be defined. That is done only when an object is created and its constructor is called.
7. Which class can have member functions without their implementation?
a) Default class
b) String class
c) Template class
d) Abstract class
View Answer

Answer: d
Explanation: Abstract classes can have member functions with no implementation, where the inheriting subclasses m ust implement those functions.
8. Which of the following describes a friend class?
a) Friend class can access all the private members of the class, of which it is a friend
b) Friend class can only access protected members of the class, of which it is a friend
c) Friend class don't have any implementation
d) Friend class can't access any data member of another class but can use it's methods
View Answer

Answer: a
Explanation: A friend class can access all the private members of another class, of which it is a friend. It is a special class provided to use when you need to reuse the data of a class but don't want that class to have those special functi ons.
9. What is the scope of a class nested inside another class?
a) Protected scope
b) Private scope
c) Global scope
d) Depends on access specifier and inheritance used
View Answer

Answer: d
Explanation: It depends on the access specifier and the type of inheritance used with the class, because if the class is inherited then the nested class can be used by subclass too, provided it's not of private type.
10. Class with main() function can be inherited.
a) True
b) False
View Answer

Answer: a
Explanation: The class containing main function can be inherited and hence the program can be executed using the d erived class names also in java.
11. Which among the following is false, for a member function of a class?

a) All member functions must be defined
b) Member functions can be defined inside or outside the class body
c) Member functions need not be declared inside the class definition
d) Member functions can be made friend to another class using the friend keyword
View Answer

Answer: c
Explanation: Member functions must be declared inside class body, though the definition can be given outside the cl ass body. There is no way to declare the member functions outside the class.
12. Which syntax for class definition is wrong?
a) class student{ };
b) student class{ };
c) class student{ public: student(int a){ } };
d) class student{ student(int a){} };
View Answer

Answer: b
Explanation: Keyword class should come first. Class name should come after keyword class. Parameterized construc tor definition depends on programmer so it can be left empty also.
13. Which of the following pairs are similar?
a) Class and object
b) Class and structure
c) Structure and object
d) Structure and functions
View Answer

Answer: b
Explanation: Class and structure are similar to each other. Only major difference is that a structure doesn't have me mber functions whereas the class can have both data members and member functions.
14. Which among the following is false for class features?
a) Classes may/may not have both data members and member functions
b) Class definition must be ended with a colon
c) Class can have only member functions with no data members
d) Class is similar to union and structures
View Answer

Answer: b
Explanation: Class definition must end with a semicolon, not colon. Class can have only member functions in its bod y with no data members.
15. Instance of which type of class can't be created?
a) Anonymous class
b) Nested class
c) Parent class
d) Abstract class
View Answer

Answer: d
Explanation: Instance of abstract class can't be created as it will not have any constructor of its own, hence while cre ating an instance of class, it can't initialize the object members. Actually the class inheriting the abstract class can ha ve its instance because it will have implementation of all members.

**********"Objects".**********

1. Which definition best describes an object?
a) Instance of a class
b) Instance of itself
c) Child of a class
d) Overview of a class
View Answer

Answer: a
Explanation: An object is instance of its class. It can be declared in the same way that a variable is declared, only thi ng is you have to use class name as the data type.
2. How many objects can be declared of a specific class in a single program?
a) 32768
b) 127
c) 1
d) As many as you want
View Answer

Answer: d
Explanation: You can create as many objects of a specific class as you want, provided enough memory is available.
3. Which among the following is false?
a) Object must be created before using members of a class
b) Memory for an object is allocated only after its constructor is called
c) Objects can't be passed by reference
d) Objects size depends on its class data members
View Answer

Answer: c
Explanation: Objects can be passed by reference. Objects can be passed by value also. If the object of a class is not c reated, we can't use members of that class.

4. Which of the following is incorrect?
a) class student{ }s;
b) class student{ }; student s;
c) class student{ }s[];
d) class student{ }; student s[5];
View Answer

Answer: c
Explanation: The array must be specified with a size. You can't declare object array, or any other linear array withou t specifying its size. It's a mandatory field.
5. The object can't be _____
a) Passed by reference
b) Passed by value
c) Passed by copy
d) Passed as function
View Answer

Answer: d
Explanation: Object can't be passed as function as it is an instance of some class, it's not a function. Object can be p assed by reference, value or copy. There is no term defined as pass as function for objects.

6. What is size of the object of following class (64 bit system)?

class student { int rollno; char name[20]; static int studentno; };
a) 20
b) 22
c) 24
d) 28
View Answer

Answer: c
Explanation: The size of any object of student class will be of size 4+20=24, because static members are not really c onsidered as property of a single object. So static variables size will not be added.
7. Functions can't return objects.
a) True
b) False
View Answer

Answer: b
Explanation: Functions can always return an object if the return type is same as that of object being returned. Care h as to be taken while writing the prototype of the function.
8. How members of an object are accessed?
a) Using dot operator/period symbol
b) Using scope resolution operator
c) Using member names directly
d) Using pointer only
View Answer

Answer: a
Explanation: Using dot operator after the name of object we can access its members. It is not necessary to use the poi nters. We can't use the names directly because it may be used outside the class.
9. If a local class is defined in a function, which of the following is true for an object of that class?
a) Object is accessible outside the function
b) Object can be declared inside any other function
c) Object can be used to call other class members
d) Object can be used/accessed/declared locally in that function
View Answer

Answer: d
Explanation: For an object which belongs to a local class, it is mandatory to declare and use the object within the fun ction because the class is accessible locally within the class only.
10. Which among the following is wrong?
a) class student{ }; student s;
b) abstract class student{ }; student s;
c) abstract class student{ }s[50000000];
d) abstract class student{ }; class toppers: public student{ }; topper t;
View Answer

Answer: b
Explanation: We can never create instance of an abstract class. Abstract classes doesn't have constructors and hence when an instance is created there is no facility to initialize its members. Option d is correct because topper class is in heriting the base abstract class student, and hence topper class object can be created easily.
11. Object declared in main() function _____
a) Can be used by any other function
b) Can be used by main() function of any other program
c) Can't be used by any other function
d) Can be accessed using scope resolution operator
View Answer

Answer: c
Explanation: The object declared in main() have local scope inside main() function only. It can't be used outside mai n() function. Scope resolution operator is used to access globally declared variables/objects.
12. When an object is returned_____
a) A temporary object is created to return the value
b) The same object used in function is used to return the value
c) The Object can be returned without creation of temporary object
d) Object are returned implicitly, we can't say how it happens inside program
View Answer

Answer: a
Explanation: A temporary object is created to return the value. It is created because the object used in function is des troyed as soon as the function is returned. The temporary variable returns the value and then gets destroyed.
13. Which among the following is correct?
a) class student{ }s1,s2; s1.student()=s2.student();
b) class student{ }s1; class topper{ }t1; s1=t1;
c) class student{ }s1,s2; s1=s2;
d) class student{ }s1; class topper{ }t1; s1.student()=s2.topper();
View Answer

Answer: c
Explanation: Only if the objects are of same class then their data can be copied from to another using assignment ope rator. This actually comes under operator overloading. Class constructors can't be assigned any explicit value as in o ption class student{ }s1; class topper{ }t1; s1=t1; and class student{ }s1; class topper{ }t1; s1.student()=s2.topper();.

14. Which among following is correct for initializing the class below?

```
class student{
int marks;
int cgpa;
public: student(int i, int  j){
marks=I;
cgpa=j
}
};
```

a) student s[3]={ s(394, 9); s(394, 9); s(394,9); };
b) student s[2]={ s(394,9), s(222,5) };
c) student s[2]={ s1(392,9), s2(222,5) };
d) student s[2]={ s[392,9], s2[222,5] };
View Answer

Answer: b
Explanation: It is the way we can initialize the data members for an object array using parameterized constructor. W e can do this to pass our own intended values to initialize the object array data.
15. Object can't be used with pointers because they belong to user defined class, and compiler can't decide the type of data may be used inside the class.
a) True
b) False
View Answer

Answer: b
Explanation: The explanation given is wrong because object can always be used with pointers like with any other var iables. Compiler doesn't have to know the structure of the class to use a pointer because the pointers only points to a memory address/stores that address.

**********"OOP Features".**********

1. Which feature of OOP indicates code reusability?
a) Encapsulation
b) Inheritance
c) Abstraction
d) Polymorphism
View Answer

Answer: b
Explanation: Inheritance indicates the code reusability. Encapsulation and abstraction are meant to hide/group data i nto one element. Polymorphism is to indicate different tasks performed by a single entity.
2. If a function can perform more than 1 type of tasks, where the function name remains same, which feature of OOP is used here?
a) Encapsulation
b) Inheritance
c) Polymorphism
d) Abstraction
View Answer

Answer: c
Explanation: For the feature given above, the OOP feature used is Polymorphism. Example of polymorphism in real life is a kid, who can be a student, a son, a brother depending on where he is.
3. If different properties and functions of a real world entity is grouped or embedded into a single element, what is it called in OOP language?
a) Inheritance
b) Polymorphism
c) Abstraction
d) Encapsulation
View Answer

Answer: d
Explanation: It is Encapsulation, which groups different properties and functions of a real world entity into single ele ment. Abstraction, on other hand, is hiding of functional or exact working of codes and showing only the things whi ch are required by the user.
Note: Join free Sanfoundry classes at Telegram or Youtube
advertisement

4. Which of the following is not a feature of pure OOP?
a) Classes must be used
b) Inheritance
c) Data may/may not be declared using object
d) Functions Overloading
View Answer

Answer: c
Explanation: Data must be declared using objects. Object usage is mandatory because it in turn calls its constructors, which in turn must have a class defined. If object is not used, it is a violation of pure OOP concept.
5. Which among the following doesn't come under OOP concept?
a) Platform independent
b) Data binding
c) Message passing

d) Data hiding
View Answer

Answer: a
Explanation: Platform independence is not feature of OOP. C++ supports OOP but it's not a platform independent la nguage. Platform independence depends on programming language.
Take Object Oriented Programming Tests Now!
6. Which feature of OOP is indicated by the following code?

```
class student{  int marks;  };
class topper:public student{  int age;  topper(int age){ this.age=age; } };
```

a) Inheritance
b) Polymorphism
c) Inheritance and polymorphism
d) Encapsulation and Inheritance
View Answer

Answer: d
Explanation: Encapsulation is indicated by use of classes. Inheritance is shown by inheriting the student class into to pper class. Polymorphism is not shown here because we have defined the constructor in the topper class but that doe sn't mean that default constructor is overloaded.
7. Which feature may be violated if we don't use classes in a program?
a) Inheritance can't be implemented
b) Object must be used is violated
c) Encapsulation only is violated
d) Basically all the features of OOP gets violated
View Answer

Answer: d
Explanation: All the features are violated because Inheritance and Encapsulation won't be implemented. Polymorphi sm and Abstraction are still possible in some cases, but the main features like data binding, object use and etc won't be used hence the use of class is must for OOP concept.
8. How many basic features of OOP are required for a programming language to be purely OOP?
a) 7
b) 6
c) 5
d) 4
View Answer

Answer: a
Explanation: There are 7 basic features that define whether a programing language is pure OOP or not. The 4 basic f eatures are inheritance, polymorphism, encapsulation and abstraction. Further, one is, object use is must, secondly, message passing and lastly, Dynamic binding.
9. The feature by which one object can interact with another object is _____
a) Data transfer
b) Data Binding
c) Message Passing
d) Message reading
View Answer

Answer: c
Explanation: The interaction between two object is called the message passing feature. Data transfer is not a feature of OOP. Also, message reading is not a feature of OOP.
10. _____ underlines the feature of Polymorphism in a class.
a) Nested class

b) Enclosing class
c) Inline function
d) Virtual Function
View Answer

Answer: d
Explanation: Virtual Functions can be defined in any class using the keyword virtual. All the classes which inherit th e class containing the virtual function, define the virtual function as required. Redefining the function on all the deriv ed classes according to class and use represents polymorphism.
11. Which feature in OOP is used to allocate additional function to a predefined operator in any language?
a) Operator Overloading
b) Function Overloading
c) Operator Overriding
d) Function Overriding
View Answer

Answer: a
Explanation: The feature is operator overloading. There is not a feature named operator overriding specifically. Func tion overloading and overriding doesn't give addition function to any operator.
12. Which among doesn't illustrates polymorphism?
a) Function overloading
b) Function overriding
c) Operator overloading
d) Virtual function
View Answer

Answer: b
Explanation: Function overriding doesn't illustrate polymorphism because the functions are actually different and th eirs scopes are different. Function and operator overloading illustrate proper polymorphism. Virtual functions show polymorphism because all the classes which inherit virtual function, define the same function in different ways.
13. Exception handling is a feature of OOP.
a) True
b) False
View Answer

Answer: a
Explanation: Exception handling is a feature of OOP as it includes classes concept in most of the cases. Also it may come handy while using inheritance.
14. Which among the following, for a pure OOP language, is true?
a) The language should follow 3 or more features of OOP
b) The language should follow at least 1 feature of OOP
c) The language must follow only 3 features of OOP
d) The language must follow all the rules of OOP
View Answer

Answer: d
Explanation: The language must follow all the rules of OOP to be called a purely OOP language. Even if a single O OP feature is not followed, then it's known to be a partially OOP language.
15. Does OOP provide better security than POP?
a) Always true for any programming language
b) May not be true with respect to all programming languages
c) It depends on type of program
d) It's vice-versa is true
View Answer

Answer: a
Explanation: It is always true as we have the facility of private and protected access specifiers. Also, only the public and global data are available globally or else the program should have proper permission to access the private data.