Kamla Nehru Mahavidyalaya

Sakkardara Square, Nagpur-24

Department of Computer Science

Question Bank

Bachelor of Vocation (B.Voc)
(Software Development)

Semester-II

Paper-II

Object Oriented Programming using 'C++'

UNIT-I

Sr. No	Questions	Correct Answer
1	Every class has at least one constructor function, even when none is declared. A) True B) False	A
2	Can constructors be overloaded? A) True B) False	A
3	What is the difference between **struct** and **class** in terms of Access Modifier? A. By default all the struct members are private while by default class members are public. B. By default all the struct members are protected while by default class members are private. C. By default all the struct members are public while by default class members are private D. By default all the struct members are public while by default class members are protected	C
4	The default access level assigned to members of a class is _____ A. Public B. Private C. Protected D. Needs to be assigned	B
5		A
6	Which of the following operators allow defining the member functions of a class outside the class? A. Scope Resolution(: :) B. (Dpt) Operator C. Conditional operator D. Dereferencing Operator(->)	A
7	Which type of class has only one unique value for all the objects of that same class? A. Static B. Friend C. Abstract D. Both A and B	A
8	What is a constructor? A. A class automatically called whenever a new object of this class is created B. A class automatically called whenever a new object of this class is destroyed C. A function automatically called whenever a new object of this class is created. D. A function automatically called whenever a new object of this class is destroyed.	C
9	Under what conditions a destructor destroys an object? A Scope of existence has finished B. Object dynamically assigned and it is released using the operator delete. C. Program terminated. D. All of the above	D
10	If a member needs to have same value for all the objects of that same class, declare the member as A. Static B. Void C. Friend D. Constant	A
11	Variables declared in the body of a particular member function are known as data members and can be used in all member	B

	A. A . True B. False	
12	In a class definition, data or functions designated private are accessible A. to any function in the program. B. only if you know the password. C. To member functions of that class. D. Only to public members of the class.	C
13	Dividing a program into functions A. is the key to object-oriented programming. B. makes the program easier to conceptualize. C. may reduce the size of the program. D. Option B and C	D
14	A function argument is A. a variable in the function that receives a value from the calling program. B. a way that functions resist accepting the calling program's values. C. a value sent to the function by the calling program. D. a value returned by the function to the calling program.	C
15	When arguments are passed by value, the function works with the original arguments in the calling program. A. True B. False	B
16	How many values can be returned from a function? A. 0 B.1 C. 2 D. 3	B

17	What is the Value of size?
	`int main() {`
	`union Data {`
	`int i_;`
	`float f_;`
	`unsigned char str[20];`
	`} data;`
	`printf("size = %d\n", sizeof(data));`
	`data.i_ = 10;`
	`data.f_ = 220.5;`
	`printf( "data.i_ : %d\n", data.i_);`
	`return 0;`
	`}`
	A. 28 B. 32 C. 20 D. 24
		C

18	What will be the output of the program?
	`#include<iostream>`
	`using namespace std;`
	`class Test {`
	`int x_;`
	`int y_;`
	`};`
	`int main() {`
	`Test t;`
	`t.x_ = 1;`
	`cout << t.x_;`
	`}`
		C

	A. 1 B. Default Value C.Compiler Error D. None of the above	

19	What is the output of the program?
	`#include<iostream>`
	`using namespace std;`
	`class Test {`
	`int x_;`
	`int y_;`
	`void setdefault() {`
	`x_ = y_ = 0;`
	`cout << x_ << " " << y_;`
	`}`
	`};`
	`int main() {`
	`Test t;`
	`t.setdefault();`
	`}`
	A. 0 0 B. x=0; y=0; C 0 D. Compilation Error	

20	Which function will change the state of the object?
	`#include<iostream>`
	`using namespace std;`
	`class Test {`
	`int x_;`
	`int y_;`
	`public:`
	`void display() { cout << x << " " << y; }`
	`void set(int m_, int n_) {`
	`x_ = m_, y_ = n_;`
	`}`
	`};`
	A. Only set() B. Only display() C. display and set both
	E. None of the above
		A

21	What will be the output of the given program? A. Compilation Error: display() cannot be accessed in application B. Compilation Error:Test class object cannot be accessed in function Demo C. Compilation Error: Variable x is private in Test D. Both A and B	C

	`#include<iostream>`
	`using namespace std;`
	`class Test {`
	`  int x;`
	`  public:`
	`  void setdefault() { x = 0; }`
	`};`
	`class Demo {`
	`  public:`
	`  display(Test t) { cout << t.x; }`
	`}`
	`int main() {`
	`  Test t;`
	`  Demo d;`
	`  d.display(t);`
	`}`	

22	Which of the following is FALSE about references in C++ A. A reference must be initialized when declared B. Once a reference is created, it cannot be later made to reference another object; it cannot be reset C. References cannot be NULL D. References cannot refer to constant value	D

24	What will be the output of following program?
	
	#include <iostream>
	using namespace std;
	
	class Test
	{
	public:
	Test() { cout <<"Hello from Test() "; }
	} a;
	
	int main()
	{
	cout <<"Main Started ";
	return 0;
	}
	A. Main Started
	B. Main Started Hello from Test()
	C. Hello from Test() Main Started
	D. Compiler Error: Global objects are not allowed
		C

25	In C++.....operator is used for Dynamic memory allocation. A. :: B. new C. Membership (Dot Operator) E. Conditional ?:	A
26	Which of the following header file includes definition of cin and cout? A. istream.h B. ostream.h C. iomanip D. iostream	D
27	cout is a/an _____	C

	A. operator B. function C. Object 4. Dariable	
28	..... is a member function with the same name as the class. A. Friend function B. Constructor C. Destructor D.None of the above	B
29	Which is not the feature of constructor? A. It cannot be inherited. B. It should be declared in Private. C. It do n1ot have return type D. All of above	B
30	Which is not type of constructor? A. Default B. Copy C. Parameterized D. None of the above	D
31	Objects are destroyed in the reverse order of its creation. A. True B. False	A
32	.... constructor is used for copying the object of same class type. A. Default B. Copy C. Parameterized D. None of the above	B
33	The function inside a class is called as ..... A. Friend function B. Member Function C. Class Function D. All of the above	B
34	How many objects can be created by a class? A. 1 B. 2 C. 3. D. Any Number .	D
35	Default return type of C++ main( ) is ..... A. float B. int C. void D. pointer	B
36	Enumerated data type is ..... A. User-defined data type B. Built-In/ Fundamental data type C. Derived data yype D. All of the above	A
37	Object-oriented programming follows Top-down approach. A. True B. False	B
38	Which type is best suited to represent the logical values? A. Integer B. boolean C. character D. All	B
39	Inline functions are A. Declared in the class defined outside the class B. Defined outside the class using keyword inline C. Defined inside the class using keyword inline D. None of the above	B
40	Which of the following functions are performed by a constructor? A. Construct a new class B. Construct a new object C. Construct a new function D. Initialize objects	D

41	What will be the ouput of following :
	#include <stdio.h>
	using namespace std;
	int array1[] = {1200, 200, 2300, 1230, 1543};
	int array2[] = {12, 14, 16, 18, 20};
	int temp, result = 0;
	int main()
	{
	for (temp = 0; temp <5; temp++) {
	result += array1[temp];
	}
		B

	for (temp = 0; temp <4; temp++) {
	result += array2[temp];
	}
	cout <<result;
	return 0;
	}
	A. 6553 B. 6533 C. 6522 C. 12200	

42	In procedural programming the focus in on …….. A. Data B. Function C. Structure D. Pointers	B
43	In object oriented programming the focus major is on …….. A. Data B. Function C. Structure D. Pointers	A
44	Which of the following data type does not return anything? A. int B. short C. long D. void	D
45	Which of the following statements is correct for a static member function? 1. It can access only other static members of its class. 2. It can be called using the class name, instead of objects A. Only 1 is correct B. Only 2 is correct C. Both 1 and 2 are correct C. Both 1 and 2 are incorrect	C
46	When a copy constructor may be called? A. When an object of the class is returned by value B. When an object of the class is passed (to a function) by value as an argument C. When an object is constructed based on another object of the same class D. All	D

47	Output of following program?
	#include<iostream>
	using namespace std;
	class Point {
	Point() { cout <<"Constructor called"; }
	};
	
	int main()
	{
	Point t1;
	return 0;
	}
	A. Compiler Error B. Run-time Error C. Constructor Called D. No Error
		A

48	What will be the output of the program? #include<iostream> using namespace std; class Point { public:	B

	Point() { cout <<"Constructor called"; }
	};
	
	int main()
	{
	Point t1,t2;
	return 0;
	}
	A. Constructor called B. Constructor called Constructor called
	C. Compiler Error D. Nothing	

49	What will be the output of the program?
	
	#include<iostream>
	using namespace std;
	
	class X
	{
	public:
	int x;
	};
	
	int main()
	{
	X a = {10};
	cout <<a.x <<" ";
	return 0;
	}
	A. 10 followed by Garbage Value B. 10 C. Compiler Error D. 0
		B

50	**The term_____means the ability to take many forms.** A. Polymorphism B. Inheritance C. Encapsulation D. Abstraction	A
51	The mechanism that binds code and data together and keeps them secure from outside world is known as A. Polymorphism B. Inheritance C. Encapsulation D. Abstraction	C
52	Member functions, when defined within the class specification A. Are always inline B. Are not inline C. Are inline by default, unless they are too big or too complicated D. Are not inline by default.	A
53	When the compiler cannot differentiate between two overloaded constructors, they are called A. Overloaded B. Destructed C. Ambiguous D. Audacious	C
54	In which case is it mandatory to provide a destructor in a class? A. Almost in every class B. Class for which two or more than two objects will be created C. Class for which copy constructor is defined D. Class whose objects will be created dynamically	D

55	Another term for_____is data hiding. A. Encapsulation B. Abstraction C. Privacy D. None of the above	A
56	The term_____refers to a way of organizing classes that share properties. A. Object-oriented B. encapsulation C. Polymorphism D. Inheritance	D
57	Encapsulation makes it easier to A reuse and modify existing modules of code B. write and read code by sharing method names. C. hide and protect data from external code. D. Both a and b.	C
58	Inheritance makes it easier to: A reuse and modify existing modules of code B. write and read code by sharing method names. C. hide and protect data from external code. D. Both a and b.	A
59	Which of the following is a C++ object? A. cin B. >> C. iostream D. read()	A
60	Which of the following is the insertion operator? A. >> B<< C. : : D. .	B
61	The term instantiation refers to the creation of: A. Class B. Object C. method D. attribute	B
62	If you declare two objects as Customer firstCust, secondCust; which of the following must be true? A. Each object's non static data members will be stored in the same memory location B. Each object will be stored in the same memory location C. Each object will have a unique memory address D. You cannot declare two objects of the same class	C
63	A static data member is initialized with a value A. within the class definition. B. outside the class definition C. when the program is executed D. never	B
64	If you want only one memory location to be reserved for a class variable, no matter how many objects are instantiated, you should declare the variable as A. static B. inline C. volatile D. dynamic	A
65	The operator that releases previously allocated memory is _____ A. release B. return C. delete D. destroy	C
66	The new operator A. returns a pointer to the variable B. creates a variable called new C. obtains memory for a new variable D. tells how much memory is available	C

67	A class defined within another class is: A. Nested class B. Local Class C. Containership D. Encapsulation	A
68	Constructors never have explicit return type A. True B. False	A
69	Can I Inherit constructor and destructor of a base class ? A. Yes B. No	B
70	Given a class named Book, which of the following is not a valid constructor? A. Book ( ) { } B. Book (Book &b){ } C. Book (Book b) { } D. Book (char* author, char* title) { }	C
71	Which value we cannot assign to reference? A. int B. float C. unsigned D. null	D

72	#include <iostream>
	using namespace std;
	int main()
	{
	int a = 9;
	int & aref = a;
	a++;
	cout << "The value of a is " << aref;
	return 0;
	}
	A. 9 B. 10 C.11 D. Error
		B

73	Identify the correct sentence regarding inequality between reference and pointer. A. we can not create the array of reference. B. we can create the array of reference. C. we can use reference to reference. D. None	A

74	#include <iostream>
	using namespace std;
	int main()
	{
	int x;
	int *p;
	x = 5;
	p = &x;
	cout << *p;
	return 0;
	}
	A. 5 B. 10 C. Memory Address D. None
		A

75		

Unit-II

Sr. No	Questions	Correct Answer
1	The keyword friend appear in A. private section of the class B. public section of the class C. Both A and B D. None of the above	C
2	It is possible to declare as a friend A. A member function B. A global function C. A class D. All of the above	D
3	A friend function to a class, C cannot access A. Private data members and member functions B. Public data members and member functions C. Protected data members and member functions D. The data members of the derived class of C	D
4	Which of the following declarations are illegal? A. void *ptr; B char *str="hello"; C. char str="hello" D. const *char c;	C
5	Identify the operator that is NOT used with pointers A & B * C -> D. >>	D
6	Which statement is true? A. A base class inherits some of the properties of a derived class. B. A base class inherits all of the properties of a derived class. C. A derived class inherits some of the properties of a base class. D. A derived class inherits all of the properties of a base class.	D
7	A pointer to the base class can hold address of A. only base class object B. only derived class object C. base class object as well as derived class object D. None of the above	C
8	When class B is inherited from class A, what is the order in which the constructers of those classes are called A. Class A first Class B next B. Class B first Class A next C. Class B's only as it is the child class D. Class A's only as it is the parent class	A
9	C++ does not supports the following A. Multiple Inheritance B. Multilevel C. Hierarchical D. None of the above	D
10	The main intention of using inheritance is …..... A. to help in converting one data type to other B. to hide the details of base class	C

	C. to extend the capabilities of base class D. to help in modular programming	

11	Is it fine to call delete twice for a pointer?
	`#include<iostream>`
	`using namespace std;`
	
	`int main()`
	`{`
	`int *ptr = new int;`
	`delete ptr;`
	`delete ptr;`
	`return 0;`
	`}`
	A. Yes B. No
		B

12	Reusability is supported by following feature A. Polymorphism B. Message passing C. Inheritance D. Operator Overloading	C
13	Base class is also known as A. Super class B. Parent Class C. Both a and b D. None of the above	C
14	Child class is also known as A. Sub class B. Derived Class C. Both a and b D. Known class	C
15	Derived class cannot access from_____base class A. Constructor B. Destructor C. Both D. Statement is false	C
16	We can derive a new class from a derived class A. True B. False	A
17	What is syntax of deriving a new class from base class is___. A. class name, new class name B. class name: access specifier class name C. new class name, base class name D. none of the above	B
18	Which constructor will initialize the base class data member A. Base class B. Derived Class C. Derived Derived D. None	A
19	Inheritance can be done using :: symbol A. True B. False	B
20	When we derived a new class using more than one class then type of inheritance is known as____. A. Multiple B. Multilevel C. Hierarchical D. Hybrid	A
21	When class B is derive from A , and class C is derived from B, this kind of inheritance is known as_____. A. Multiple B. Multilevel C. Hierarchical D. Hybrid	B
22	class A: public B, public C is a type of inheritance A. Multiple B. Multilevel C. Hierarchical D. Hybrid	A
23	Virtual base class is used to_____. A. to perform operator overloading B. to perform function overloading C. to remove ambiguity in hybrid inheritance	C

	D. all of the above	
24	Make a correct sequence of a statement i)destructor of derived class is called ii)destructor of base class is called iii)constructor of derived class is called iv)constructor of base class is called A. i,ii,iv,iii B. iv,iii,ii,I C. iv,iii,i,ii D. i,ii,iii,iv	C
25	If a base class contains a member function basefunc(), and a derived class does not contain a function with this name, can an object of the derived class access basefunc()? A. Yes B. No	A
26	Assume a class **Derv** that is privately derived from class **Base**. An object of class **Derv** located in **main()** can access **A.** public members of **Derv.** **B.** protected members of **Derv**. **C.** private members of **Derv**. **D.** public members of **Base**.	A

27	What is the output of the program?
	
	#include <iostream>
	#include <string>
	using namespace std;
	class Department {
	public:
	string dept;
	Department(string d):dept(d) { }
	void getDeptName() { cout <<dept; }
	};
	class Student : private Department {
	public:
	string name;
	Student(string n = "Not entered", string d = "ATDC") :
	name(n), Department(d) { }
	using Department::getDeptName;
	};
	int main() {
	Student s("CSE");
	s.getDeptName();
	return 0;
	}
	A. CSE B. ATDC C. Not entered D. Compilation Error
		B

28	Inheritance can be done using :: symbol A. True B. False	B
29	Identify the lines on which the compiler will report an error. #include <iostream>// ---1	D

	using namespace std; // ---2
	class Base { // ---3
	int var_; // ---4
	public: // ---5
	Base():var_(0){} // ---6
	}; // ---7
	class Derived: public Base { public: // ---8
	int varD_; // ---9
	void print () { cout <<var_; } // ---10
	}; // ---11
	int main() { // ---12
	Derived d; // ---13
	d.var_ = 1; // ---14
	d.varD_ = 1; // ---15
	cout <<d.var_ <<""<<d.varD_; // ---16
	return 0; // ---17
	} // ---18
	A. 6, 10, 14, 15 B. 6, 15 C. 6, 14, 16 D. 10, 14, 16	

30	#include <iostream>
	using namespace std;
	class Base { public:
	int var_;
	void func(int){}
	};
	class Derived: public Base { public:
	int varD_;
	void func(int){}2
	};
	int main() {
	Derived d;
	d.func(1);
	return 0;
	}
	Which of the following function will be invoked by d.func(1)?
	A. Base::func(int)
	B. Derived::func(int)
	C. Compilation Error
	D. None of the above
		B

31	What is the output of the following program? #include<iostream> #include<string> using namespace std; class Base { public: void func_f1(int i) { cout <<"In base func_f1 "; }	B

	void func_f2(int i) { cout <<"In base func_f2 "; }
	};
	class Derived: public Base {
	public:
	void func_f1(int i) { cout <<"In derived func_f1 "; }
	void func_f1(string s) { cout <<"func_f1 string "; }
	void func_f3(int i) { cout <<"In derived func_f3 "; }
	};
	int main() {
	Base b;
	Derived d;
	d.func_f1(3);
	d.func_f1("Blue");
	d.func_f3(3);
	d.func_f2(3);
	return 0;
	}
	A. Compilation Error: Cannot add new parameters to func_f1
	B. In derived func_f1 func_f1 string In derived func_f3 In base func_f2
	C. In base func_f2 func_f1 string In derived func_f3 In derived func_f1
	D. Compilation Error: Cannot define func_f3 containing same parameter type as func_f1	

32	What is the output of the following program? {Assume size of int as 4}
	
	#include<iostream>
	using namespace std;
	class base {
	int data;
	};
	class derived1: public base { };
	class derived2: public derived1 { };
	int main() {
	cout <<sizeof(derived2);
	return 0;
	}
		A

33	What will be the output of the following program?
	
	#include<iostream>
	using namespace std;
	class Base { public:
	Base() { cout <<"Base Ctor "; }
	~Base() { cout <<"Base Dtor "; }
	};
		D

	class Derived: public Base { public:
	Derived() { cout <<"Derived Ctor "; }
	~Derived() { cout <<"Derived Dtor "; }
	};
	int main() {
	Derived d1;
	{
	Base b1;
	}
	return 0;
	}
	A. Base Ctor Derived Ctor Base Ctor Base Dtor Base Dtor Derived Dtor
	B. Derived Ctor Base Ctor Base Ctor Base Dtor Derived Dtor Base Dtor
	C. Derived Ctor Base Ctor Base Dtor Derived Dtor
	D. Base Ctor Derived Ctor Base Ctor Base Dtor Derived Dtor Base Dtor	

34	What will be the output of the program?
	
	#include <iostream>
	using namespace std;
	class F1 {
	public:
	F1() { cout <<"F1 ctor "; }
	~F1() { cout <<"F1 dtor "; }
	};
	class F2 : public F1 {
	public:
	F2() { cout <<"F2 ctor "; }
	~F2() { cout <<"F2 dtor "; }
	};
	class F3 : public F1 {
	const F2 &f2;
	public:
	F3() : f2(*new F2) { cout <<"F3 ctor "; }
	~F3() { cout <<"F3 dtor "; }
	};
	int main() {
	F3 f3;
	return 0;
	}
	A. F1 ctor F2 ctor F3 ctor F3 dtor F2 dtor F1 dtor
	B. F1 ctor F1 ctor F2 ctor F3 ctor F3 dtor F1 dtor
	C. F1 ctor F3 ctor F3 dtor F1 dtor
	D. F1 ctor F1 ctor F2 ctor F3 ctor F3 dtor F2 dtor F1 dtor F1 dtor
		B

35	What will be the output of the program?
	
	#include <iostream>
	using namespace std;
	class Room {
	int number;
	public:
	Room(int num = 0): number(num) { }
	void dimension() { cout <<number <<"Rooms "; }
	};
	class Building {
	public:
	Building() : ro(100) { }
	void Build() { ro.dimension(); }
	private:
	Room ro;
	};
	int main() {
	Building B;
	B.Build();
	return 0;
	}
	A. 0 Rooms B. 100 Rooms C. Compilation Error: ro is private D. None of the mentioned
		B

36	When deriving a class from with **protected** inheritance, **public** members of the base class become_____members of the derived class, and **protected** members of the base class become _____members of the derived class. A. protected, protected. B. public, private C. private, private D. Private, protected	A
37	When deriving a class with **public** inheritance, **public** members of the base class become_____members of the derived class, and **protected** members of the base class become _____members of the derived class. A. private, private B. public, protected. C. protected, protected D. Private, protected	B
38	Which of the following is true about this pointer? A. It is passed as a hidden argument to all function calls B. It is passed as a hidden argument to all non-static function calls C. It is passed as a hidden argument to all static functions D. None of the above	B
39	The inheritance is described as a_____relationship A. has a B. is a C. association D. None of these	B
40	When the object of derived class expire, first the_____is invoked followed by the_____. A. derived class constructor, base class destructor	B

	B. derived class destructor , base class destructor C. base class destructor , derived class destructor D. none of these	
41	If class A is a friend class of class B, if class B is friend class of class C then_____ A. class C is friend class of A B. class A is friend class of C C. class A and class C do not have any friendship relation. D. none of these	C
42	When base class pointer points to derived class object_____ A. it can access only base class members B. it can access only derived class members C. both base class &derived class members D. None of these	A

43	What will happen on execution of the following code ?
	Class base
	{
	};class derived: protected base
	{
	};
	A. It will not compile as the class body of the base class is not defined
	B. It will not compile as the class body of the derived class is not defined
	C. It will compile successfully
	D. The compilation of above code is dependent upon the type of data provided to it
		C

44	How many types of inheritance are there A. 2 B. 3 C. 4 D.5	D
45	Suppose class derived is derived from a class Base. Both the classes contain the Function name display() that take no argument. What will be the statement in the class derived which will called the display function of base class A. display() B. Base:display() C. Base ::display() D. Cannot make such a call	C
46	Suppose class derived is derived from a class Base privately. The object of class Derived is located in main() can access_____. A. public members of base B. private members of base C. protected members of base D. public members of derived	D
47	Which is the correct class definition for class C ,Which inherits from A &B classes A. Class C:A,B B. Class C::A,B C. Class C:public A,public B D. Class C::public A,public B	C
48	What does the following statement mean? int (*fp)(char*)	C

	A. pointer to a pointer B. pointer to an array of chars C. pointer to function taking a char* argument and returns an int D. function taking a char* argument and returning a pointer to int	
49	The operator used for dereferencing or indirection is _____ A. * B. & C. -> D. ->>	A
50	Choose the right option string* x, y; A. x is a pointer to a string, y is a string B. y is a pointer to a string, x is a string C. both x and y are pointer to string types D. both x and y are string types	A

51	#include <iostream>
	using namespace std;
	int main()
	{
	int a = 5, b = 10, c = 15;
	int *arr[ ] = {&a, &b, &c};
	cout <<arr[1];
	return 0;
	}
	A. 10 B. 15 C. 20 D. Random number
		D

52	The correct statement for a function that takes pointer to a float, a pointer to a pointer to a char and returns a pointer to a pointer to a integer is A. int **fun(float**, char**) B. int *fun(float*, char*) C. int ***fun(float*, char**) D. int ***fun(*float, **char)	C

53	#include <iostream>
	using namespace std;
	int main()
	{
	char arr[20];
	int i;
	for(i = 0; i < 10; i++)
	*(arr + i) = 65 + i;
	*(arr + i) = '\0';
	cout << arr;
	return(0);
	}
	A. ABCDEFGHIJ
	B. AAAAAAAAA
	C. JJJJJJJJJJ
	D. None
		A

54	#include <iostream>	A

	using namespace std;
	int main()
	{
	char *ptr;
	char Str[] = "abcdefg";
	ptr = Str;
	ptr += 5;
	cout << ptr;
	return 0;
	}
	A. fg B. cdef C. defg D. abcd	

55	#include <iostream>
	using namespace std;
	class Box
	{
	double width;
	public:
	friend void printWidth(Box box);
	void setWidth(double wid);
	};
	void Box::setWidth(double wid)
	{
	width = wid;
	}
	void printWidth(Box box)
	{
	box.width = box.width * 2;
	cout << "Width of box : " << box.width << endl;
	}
	int main( )
	{
	Box box;
	box.setWidth(10.0);
	printWidth(box);
	return 0;
	}
	A. Width of box : 40 B. Width of box : 5 C. Width of box : 10
	D. Width of box : 20
		D

56	#include <iostream> using namespace std; class sample { private: int a, b; public: void test()	C

	{
	a = 100;
	b = 200;
	}
	friend int compute(sample e1);
	};
	int compute(sample e1)
	{
	return int(e1.a + e1.b) - 5;
	}
	int main()
	{
	sample e;
	e.test();
	cout << compute(e);
	return 0;
	}
	A. 100 B. 200 C. 295 D. 395	

57	#include <iostream>
	using namespace std;
	class base
	{
	int val1, val2;
	public:
	int get()
	{
	val1 = 100;
	val2 = 300;
	}
	friend float mean(base ob);
	};
	float mean(base ob)
	{
	return float(ob.val1 + ob.val2) / 2;
	}
	int main()
	{
	base obj;
	obj.get();
	cout << mean(obj);
	return 0;
	}
	A. 200 B. 150 C. 100 D. 250
		A

58	To which does the function pointer point to? A. Function B. Variable C. Constant D. Object	A
59	What we will not do with function pointers?	C

	A. Allocation B. Deallocation C. Both A & B. D. Do Both A &B	

60	#include <iostream>
	using namespace std;
	int add(int first, int second)
	{
	return first + second + 15;
	}
	int operation(int first, int second, int (*functocall)(int, int))
	{
	return (*functocall)(first, second);
	}
	int main()
	{
	int a;
	int (*plus)(int, int) = add;
	a = operation(15, 10, plus);
	cout << a;
	return 0;
	}
	A. 25 B. 36 C. 40 D. 45
		C

61	#include <iostream>
	using namespace std;
	void func(int x)
	{
	cout << x ;
	}
	int main()
	{
	void (*n)(int);
	n = &func;
	(*n)(2);
	n(2);
	return 0;
	}
	A. 2 B. 21 C. 22 D. 23
		C

62	#include <iostream>
	using namespace std;
	int n(char, int);
	int (*p) (char, int) = n;
	int main()
	{
	(*p)('d', 9);
	p(10, 9);
	return 0;
	}
		A

	int n(char c, int i)
	{
	cout << c << i;
	return 0;
	}
	A. d9 9 B. d9 d9 C. d9 D. Compile time error	

63	#include <iostream>
	using namespace std;
	int func (int a, int b)
	{
	cout << a;
	cout << b;
	return 0;
	}
	int main(void)
	{
	int(*ptr)(char, int);
	ptr = func;
	func(2, 3);
	ptr(2, 3);
	return 0;
	}
	A. 2323 B. 232 C. 23 D. Compiler Error
		D

64	What is meaning of following declaration? int(*ptr[5])(); A. ptr is pointer to function. B. ptr is array of pointer to function. C. ptr is pointer to such function which return type is array. D. ptr is pointer to array of function.	A
65	Void pointer can point to which type of objects? A. int B> float C. char D. All	D
66	What does the following statement mean? int (*fp)(char*) A. pointer to a pointer B. pointer to an array of chars C. pointer to function taking a char* argument and returns an int D. function taking a char* argument and returning a pointer to int	C

67	#include <iostream>
	using namespace std;
	int main()
	{
	int a[2][4] = {3, 6, 9, 12, 15, 18, 21, 24};
	cout << *(a[1] + 2) << *(*(a + 1) + 2) << 2[1[a]];
	return 0;
	}
		B

	A. 15 18 21 B. 21 21 21 C. 24 24 24 D. Compiler Error	

68	#include <iostream>
	using namespace std;
	int main()
	{
	int arr[] = {4, 5, 6, 7};
	int *p = (arr + 1);
	cout << *p;
	return 0;
	}
	A. 4 B. 5 C. 6 D. 7
		B

69	#include <iostream>
	using namespace std;
	int main()
	{
	int arr[] = {4, 5, 6, 7};
	int *p = (arr + 1);
	cout << arr;
	return 0;
	}
	A. 4 B. 5. C. Address of arr D. 6
		C

70	#include <iostream>
	using namespace std;
	int main ()
	{
	int numbers[5];
	int * p;
	p = numbers; *p = 10;
	p++; *p = 20;
	p = &numbers[2]; *p = 30;
	p = numbers + 3; *p = 40;
	p = numbers; *(p + 4) = 50;
	for (int n = 0; n < 5; n++)
	cout << numbers[n] << ",";
	return 0;
	}
	A. 10,20,30,40,50 B. 1020304050 C. Compile time error D. Runtime Error
		A

71	#include <iostream>
	using namespace std;
	int main()
	{
	int arr[] = {4, 5, 6, 7};
	int *p = (arr + 1);
	cout << *arr + 9;
		C

	return 0; } A. 12 B. 5 C. 13 D. Error	
72	A void pointer cannot point to which of these? A. methods in c++ B. class member in c++ C. all of the mentioned D. None	C

73	#include <iostream>
	using namespace std;
	int main()
	{
	int *p;
	void *vp;
	if (vp == p);
	cout << "equal";
	return 0;
	}
	A. equal B. No output C. Error D. Exception
		A

74	#include <iostream>
	using namespace std;
	int main()
	{
	int n = 5;
	void *p = &n;
	int *pi = static_cast<int*>(p);
	cout << *pi << endl;
	return 0;
	}
	A. 5 B. 6 C. Compile Time Error D. Run Time Error
		A

75	#include <iostream>
	using namespace std;
	int main()
	{
	int a = 5, c;
	void *p = &a;
	double b = 3.14;
	p = &b;
	c = a + b;
	cout << c << '\n' << p;
	return 0;
	}
	A. 8, memory address B. 8.14 C. memory address D. None
		A

76	What we can't do on a void pointer? A. Pointer Arithmetic B. Pointer Functions C. Both D. None	A
77	#include <iostream>	A

	using namespace std;
	int main()
	{
	double arr[] = {5.0, 6.0, 7.0, 8.0};
	double *p = (arr+2);
	cout << *p << endl;
	cout << arr << endl;
	cout << *(arr+3) << endl;
	cout << *(arr) << endl;
	cout << *arr+9 << endl;
	return 0;
	}
	A. 7
	0xbf99fc98
	8
	5
	14
	B. 7
	8
	0xbf99fc98
	5
	14
	C. 0xbf99fc98 D. None	

78	#include <iostream>
	using namespace std;
	class Foo
	{
	public:
	Foo(int i = 0){ _i = i;}
	void f()
	{
	cout << "Executed"<<endl;
	}
	private:
	int _i;
	};
	int main()
	{
	Foo *p = 0;
	p -> f();
	}
	A. Executed B. Copiler Error C. Run time Error D. None
		A

79	A pointer to a base class can point to objects of a derived class. A. TRUE B. FALSE	A
80	What does the **this**pointer point to?	B

	A. Data member of class B. the object of which the function using it is a member C. Member function D. Base class	
81	A pointer is A. the address of a variable. B. an indication of the variable to be accessed next. C. a variable for storing addresses. D. the data type of an address variable.	C
82	The expression ***test** can be said to A. refer to the contents of test. B. dereference test. C. refer to the value of the variable pointed to by test. D. All of the above	D
83	Definition for an array **arr**of 8 pointers that point to variables of type **float** is A. *float arr[8] B. float* arr[8]; C. float pointer[8] D. int *ary[8]	B

1. Which allows you to create a derived class that inherits properties from more than one base class?

A. Multilevel inheritance

B. Multiple inheritance

C. Hybrid Inheritance

D. Hierarchical Inheritance

2. Which feature in OOP allows reusing code?

A) Polymorphism

B) Inheritance

C) Encapsulation

D) Data hiding

3. A function that changes the state of the cout object is called a(n) _____

A. member B. adjuster

C. manipulator D. operator

4. What does C++ append to the end of a string literal constant?

A. a space

B. a number sign (#)

C. an asterisk (*)

D. a null character

5. An array element is accessed using

A. a first-in-first-out approach

B. the dot operator

C. a member name

D. an index number

6. To hide a data member from the program, you must declare the data member in the _____ section of the class

A. concealed B. confidential

C. hidden D. private

E. restricted

7. External documentation includes

A. a printout of the program's code

B. flowcharts

C. IPO charts

D. pseudo code

E. All of the above

8. The function whose prototype is void getData(Item *thing); receives

A. a pointer to a structure

B. a reference to a structure

C. a copy of a structure

D. nothing

9. Null character needs a space of

A. zero bytes

B. one byte

C. three bytes

D. four bytes

10. The number of structures than can be declared in a single statement is

A. one B. two

C. three D. unlimited

11. Which of the following formulas can be used to generate random integers between 1 and 10?

A. 1 + rand() % (10 - 1 + 1)

B. 1 + (10 - 1 + 1) % rand()

C. 10 + rand() % (10 - 1 + 1)

D. 10 + rand() % (10 + 1)

12. Format flags may be combined using the _____

A. bitwise OR operator (|)

B. logical OR operator (||)

C. bitwise AND operator (&)

D. logical AND operator (&&)

13. Which of the following will store the number 320000 as a Float number?

A. counPop = (float) 3.2e5;

B. counPop = (float) 3.2e6;

C. counPop = (float) .32e5;

D. counPop = (float) .32e7;

14. The arguments that determine the state of the cout object are called

A. classes

B. manipulators

C. format flags or state flags

D. state controllers

15. The following statement where T is true and F is false T&&T||F&&T

A. is true

B. is false

C. is wrong

D. not applicable in C language

16. Which of the following statements declares a variable that can contain a decimal number?

A. dec payRate;

B. dec hourlyPay

C. float payRate

D. float hourlyPay;

17. The statement int num[2][3]={ {1,2}, {3,4}, {5, 6} };

A. assigns a value 2 to num[1][2]

B. assigns a value 4 to num[1][2]

C. gives an error message

D. assigns a value 3 to num[1][2]

18. A program will have one function prototype for each function defined in the programmer-defined section of the program. (Assume that the programmer-defined section is located below the main function.)

A. true B. false

19. The standard input stream, which refers to the keyboard, is called

A. cin

B. cout

C. stin

D. stout

20. Elements in an array are identified by a unique _____

A. data type

B. order

C. subscript

D. symbol

21. The statement fwrite ((char*)&objl, sizeof(objl));

A. writes the member functions of objl to fl

B. writes the data in objl to fl

C. writes the member functions and me data of obj 1 to fl

D. writes the address of objl to fl

22. The body of a C++ function is surrounded by _____

A. parentheses

B. angle brackets

C. curly brackets

D. square brackets

23. Which of the following type casts will convert an Integer variable named amount to a Double type?

A. (double) amount

B. (int to double) amount

C. int to double(amount)

D. int (amount) to double

24. The loosest type of coupling is

A. data coupling

B. control coupling

C. external coupling

D. pathological coupling

25. Which of the following is a string literal constant?

A. "Visual C++"

B. "137.45"

C. "A"

D. "2,365"

E. All of the above

26. Which of the following, if any, are valid names for variables?

A. class

B. friend

C. #OnHand

D. void

E. None of the above is valid names for variables

27. You have assigned the address of Value to the pointer P, Which statement will display the value stored in Value?

A. cout<<P; B. cout<<*Value;

C. cout<<&P; D. cout<<*<P;

28. The void specifier is used if a function does not have return type.

a. True

b. False

29. You must specify void in parameters if a function does not have any arguments.

a. True

b. False

30. Type specifier is optional when declaring a function

a. True

b. False

31. A pointer to a block of memory is effectively same as an array

A. True B. False

32. Does this mentioning array name gives the base address in all the contexts?

A. Yes B. No

33. Is there any difference int the following declarations?

int fun(int arr[]);

int fun(int arr[2]);

A. Yes B. No

34. Are the expressions arr and &arr same for an array of 10 integers?

A. Yes B. No

35. The keyword used to transfer control from a function back to the calling function is

A. switch B. goto

C. go back D. return

Answer:-

1. **B**
2. **B**
3. **C**
4. **D**
5. **D**

6. D
7. A
8. A
9. B
10. D
11. A
12. A
13. A
14. C
15. A
16. D
17. C
18. A
19. A
20. C
21. B
22. C
23. A
24. A
25. E
26. E
27. D
28. A
29. B
30. B
31. A
32. B
33. B
34. B
35. D