	Assignment - 2
	choose the correct option
1	Local variables are stored in an area called
Sol	cd) Stack
	(1,0000 +1,0000 (1,01)
	Choose the correct option?
	#include Using namespace std;
	class Derived: public Base & };
1840	int main ()
	& Base *bp = new Derived; Derived *dp = new Base;
1303	12
Solt	CC) Compiler Error in line "Derived * dp = new Base ?
(19)	the first form the second the sec
3)	When the inheritance is private, the private methods
	in base class are in the derived class
	Cin C++).
201>	(a) Inaccessible
4)	which of the following is true?  (a) The number of times destructor is called defends  on Number of objects created.
2017	ca) The number of times destructor is called defends
	on Number of objects created.
5)	State True or False
	Type conversion is automatic whereas type casting
Sal	is explicit.
NO1->	True
<b>C</b> Scan	ned with CamScanner

	Page No.
	Date /
	Clark as a fuho a restion?
-	Short answer type question?
1	Explain about new and delete Keywords with code.
8019	The new oberator denotes a request for memory
90	allocation on the free store. If sufficient memory
	is available, new oberator initializes the memory
	and netwins the address of the newly allocated
	and initialized memory to the pointer variable.
	Syntax:
	pointer-variable = new data-type;
	Example:
	int * p = new int (25); float * q = new float (23-45);
4	float *q = new float (23-45);
	. Valid mari Pinki kanderai
	(b) & The delete operator is used to deallocate
200	the memory. User has privilege to deallocate the
	created pointer variable by this delete operator.
- 43000	
9-1	Syntax >
	delete pointer_variable;
	delete pointer_variable; Syntax to delete the block of allocated memory:
-	
	delete [] pointer_variable;
1.3	Crampile:
	delete p; delete [79:
	delete[]q:
Ce	and with Component
ar	nned with CamScanner

_=	Date /
2.	What are constructors? Why they are neguined?
17.8	What are constructors? Why they are nequired? Explain different types of constructors with suitable example.
Sol->	Constructor -
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	A constructor is a member function of a dell
	The state of the s
	constructor is automatically called when
	special member function of the class.
	There are three types of constructors:  1) Default constructors: Default constructor which doesn't take any argument. It has no barrameters.
	1) Default constructors - Default constructor
	which doesn't take any argument. It has
	15 pt 30 7 - 1 0 05
	Example:
- 1	Epublic:
	inta, b;
	construct ()
4	$\delta a = 10$ ;
	b = 20;
	3
is Ab.	43
	int main()
	(Another )
6 Busi	construct c;
	construct c; cont<<66 a; "><< c.a<< end; cont<< 66 b; "><< C.b; return 1;
	return 1:
Scanne	Cam Scanner 4

Date 2) Parameterized constuctors: It is possible to pass arguments to constituetors. Typically, these arguments help initialize an object when it is created. To create a parameterized constructor, simply add parameters to it the way you would to any other function when you define the constructor's body use the parameters to initialize the object. Example: class Point & private int x , y ; Point (intx1, inty1) P X = X1: y= y1; intgetX() freturn x; ? intgety() Fretwiny; int main () cout < 66 P1. x = 12 < p1. getx() < 86 P1. y = 72 < p1. getx() 3). Copy constructor: A copy constructor is a member function which initializes an object using another ned with CamScanner

12.50	Date /
	Example:
	class point
	Sprivate:
	double x, y;
	public:
+ NE	point (double px, double py)
100	$\sum_{y} \sum_{y} \sum_{y$
	4. 9
	Jo
	int main (void)
	point a [10];  point b @ = point (5,8);
100	7.
	9
3.	Explain the difference blw object oriented and
	Explain the difference blw object oriented and procedural programming language in detail
dol>	WILLIAM TO PROPER SONT
	Object Oriented Programming Procedural Oriented Programming
	T ( 1 a santo ( and ) is T and a fine
(1)	In object oriented programming (i) In procedural programming,
	small parts called objects small parts scalled functions
	SWIGHT POUS CHARLO OFFICES, STRAND POUS SCARCE FUNIT WORK
_ cii)	Object oriented programming (ii) Procedural programming
B. All	follows bottom up approach follows top down approach
1-1-1-1-	
رأأن	Object oriented programming (iii) There is no access
	have acess specifiers like specifier in procedural
36	private, public protected programming
Scanne	will tamscanner to the same of

	Date ,
Ĩ.	iv Object oriented programming (iv) Procedural programming properly provides data hiding soit does not have any properly as more secure.  Less secure.
C	iv object oriented programming city reactions any property
3	provides data hiding soit does vor hiding data so itic
	as more secure.
	188 882006
	Overloading is possible in cv. In procedural programming object oriented programm overloading is not possible
CV	Overloading is possible in comprocesses is not possible
	Overloading is possible in cv) In procedural programming object oriented programm overloading is not possible
	ing.
	1 21 1 22 22 2 2 2 2 2 2 2 2 2 2 2 2 2
CVI	) Object oriented programming (vi) Procedural programming is based on neal world is based on unreal world
	is based on sical world. Is based or winds we say
	TOWN TO BELLEVILLE TO FAR TO ANY
CVII	etc. Examples: - C++, Java, fython (vii) Examples: - C, FOR TRAN, foscal etc.
	etc.
	The state of the s
	Long answer Type question.
1.00	
a)	Explain the type of polymorphism with code, In C++ polymorphism is mainly divided into two types:
Soly	In C++ polymorphism is mainly divided
P. I	into two types:
Later, F	(i) Compile time polymorphism
	ciis Runtime polymorphism
3- 11	The graph of the same of the s
A Later I	· Compile time polymorphism: This type of bolymorphism
	is achieved by function overlanding or spector
	overloading of operator
1. 40	Street Control of the
-	> function overloading: - When there are multiple with
	Same name but different parameters the these
	same name but different parameters the these
- Scallin	ou with outhocaling

	Page No.  Date /
_ =	
	functions are said to be ovorloaded - functions can
9 7	be overloaded by change in number of arguments
-	or/and change in type arguments.
15	Example:
20	# include < iostream>
g Sq.	using namespace std;
	void print (inti) of cout <2 66 Here is int 97 << i << endlight
	Reserved the second of the sixther will be a
1	Void print (double F) &
3.3.	coutece Here is float "<< f << endl;
	4
	void print (char const *C) {  cont < 66 Here is char * < < < < endl;
	cout << 60 Here is char << << endl;
	(10)
	int main () {
	pount (10);
	print (10.10);
	pount (66 ten 19);
	netwin 0;
1	J
1	The second secon
	the state of the s
Y	the same of the same of the same of
	The second secon

	to law ide of time to
100	-> Operator Overloading: Ct+ also provide option to
70.	-> Operator overloading-Ctt as produced in make overload operators-for example, we can make
. 60	1 ( ) 10 (
	al as a toby Mala talk is TA add (WOUD)
	as acre about the star of the heart of the heart of the heart of the start of the heart of the h
	adds them and when placed between string operans,
	concatenates them
	forexample # include < iostream
	using namespace 5td;
	class Complex S
	private:
	int real, imag;
	public:
	Complex (intr=0, inti=0) Sreal=v; imag=i; }
	Complex operator + (Complex const & obj) 5
	Complex ores;
	res-real = real + obj-real;
	res. imag = imag + obj-imag;
	return pres;
	2
	void print () & cout << real < 266 + i 12 cinaquend); }
	2. Cour - real 200 + i cinsagezend);
	int main()
200	& complex of (10 5)
	(6)mplex (1(10,5), (2(2,4));
	(3 alixed)
	2 CS-prilate();
	g
CS cann	ed with CamScanner

100	
	2 8 + 2 2 - 14
	solymorphism: This type of or
	2. Runtime polymorphism: This type of polymorphism is achieved by Function Overriding.
	hand occurs when a derved class has a
	activities the same las
	The state of the state of the state of the
	3 - 0 - 0 - W - ( - ( ) )
1000	Example: # include < bits/stdc++.h>
-32/	using namespace std:
- 10	class base
100	¿public:
	Viltual void point ()
	Scout < 2 66 print base class '2< end; }
	Void Show()
in the second	& cout << 66 show base class"> < endl; 3
	7,
	class derived: public base
The second	& public:
	void print ()
	Scout << 66 point derived class 19 << endl; 3
	void show()
	Scout < 2 61 show derived class 19 < endl; 3
	7
W	int main()
11	E base * bptn; derivedd; bptr = 4d;
	derived d:
5.4	botr = 4 d:
	betr -> print():
- 1	bptr -> chauly.
	bptr -> print(); bptr -> show(); med with Camsgan Detwin O;
200	7.

	Date /
Solco	# include < iosteream>
	#include <strings< td=""></strings<>
	using namespace Std;
	class member s
	char name [20], address [40];
	double numbe;
	intage;
	publica management of the management
	int salary
	void input ()
•	of cout < endl;
	contex 66 Name: "> < endl;
17	
	cin get line (name, 20); cout << 66 Age: 11 < cendl;
3.	cout << 16 Phone Number: 39 << endl;
	Cout << 16 Phone Number: 39 << endl;
	cin>> number:
	cout << " Address: " << endl;
11 4	cin-getline (address, 40); cont < 66 Salary: 99 < endl;
1/4	cont < 66 Salary: 97 << endl;
- 3	cin>> salary;
	3
45	void display()
	? cout <= Endl;
1	cout <2"Name: "><< name<< endl;
- 57	cout << " Age: " << age << endl;
	cout << 66 Phone Number: ">< numbe << end;
100	cout << "Address: " << address << endl;
A STORY	cont << "Salary: " << salary << endl;
Cignann	d with CarnScanner

	char specialization [20], department [20];
	char Specialization [20], department [20];
	public:
	Void input ()
	Scout <2 16 In It Enter Employee Details It In 99;
_	member: input();
	cout < 6 Specialization: 19 exendl;
	cout < 66 Department: 12 × endl;
	cout < 66 Department: 3/2 cendl;
_	Cin-getline (department, 20);
	9
	void display ()
	Scout < 2 % In It Displaying Employee Details It In ";
-	
	course. Specialization; ">22 Specialization centle
	contes (6 Specialization; "Lespecialization end; contes (6 Department: 11 ex department exend);
	Void printsalory ()
	Scout < "In Salary of the member is: "> (salary exendl; )
	class improperty and the
	char specialization [20], department [20]
	char specialization [20], department [20];
	Void input()
	g coult <26 In It Enter Manager Details It In?
	Cont 22 66 She cializ dies 19
	cin getline (checialization : "Ezenalia
	contestine (specialization: 19 = cendlication, 20); contestine (specialization, 20); contestine (department: 1) & endlication (department) (department)
	Cin getline (department, 20); 6
CS ca	nned with CamScanner

	Page No.
	Date /
	void display ()
	Sout cell in It Displaying 1-100-5
	member: display()
	Void display ()  Sout <2619n   t Displaying Manager Details   t   n'!;  member: display ()  cout <26 Specialization: " <2 specialization <2 endl;  cout <26 Department: " <2 department <2 endl;
	cout < 6 Department = 9122 department
	Void printSalary()  § cout << 66 \n Salary of the memberis: ">Cout << 66 \n Salary of the memberis: ">Cout << endl;
	5 suit 66 1 salary of the member 13:
	cout countie
	2
	1 . 4 .
	int main ()
	\$
	employee e;
An and a second	marager m; e.input();
	e-input();
	m-input();
	e-display();
	e new 4 Colory ()
45.5	m. display (); m. print Salary ();
	m phint School / )
	b pard sawy ();
	J
201.5°	
CS canned	with CamScanner