MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Object Oriented Programming **Subject Code: 17432**

Important Instructions to examiners:

1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills).
4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q. No.	Sub Q.N.	Answer	Marking Scheme
1.	**(A)** **(a)** Ans.	**Attempt any SIX of the following:** **Which are the input-output operator in C++? Give suitable example.** 1. Input operator: >> extraction or get from operator *Example*: cin>> number; 2. Output operator: << insertion or put to operator *Example*: cout<<number;	**12** **2M** ***List of two operators-1M Example of each ½ M***
	(b) Ans.	**Give significance of '&' and '*' operators.** **Address operator:-&** It is used to retrieve address of a variable. With address operator address of a variable can be stored in pointer variable. **Pointer operator:- * operator** It is used to declare a pointer variable. Also used as 'value at' operator to read value stored inside the address pointed by pointer.	**2M** ***Significance of each 1M***

```
ptr1->accept();
ptr1++;
}
ptr1=ptr;
for(i=0;i<5;i++)
{
ptr1->display();
ptr1++;
}
getch();
}
```

3. **(a)** Ans.

Attempt any FOUR of the following: **16**

Explain the structure of C++ program with suitable example. **4M**

General C++ program has following structure.

INCLUDE HEADER FILES
DECLARE CLASS
DEFINE MEMBER FUNCTIONS
DEFINE MAIN FUNCTION

Structure 1M

Description:-

1. Include header files

In this section a programmer include all header files which are require to execute given program. The most important file is *iostream.h* header file. This file defines most of the C++statements like *cout* and *cin*. Without this file one cannot load C++ program.

Description 2M

2. Declare Class

In this section a programmer declares all classes which are necessary for given program. The programmer uses general syntax of creating class.

3. Define Member Functions

This section allows programmer to design member functions of a class. The programmer can have inside declaration of a function or outside declaration of a function.

4. Define Main Functions

This section the programmer creates object and call various functions writer within various class.

Example:

```
#include<iostream.h.
#include<conio.h>
```

```
        int a,b,temp;
        float c,d;
        cout<<"Enter value for a & b="<<endl;
        cin>>a>>b;
        swap(a,b);
        cout<<"Enter value for c & d="<<endl;
        cin>>c>>d;
        swap(c,d);
        getch();
}
```

(e) Ans.

What is 'this' pointer? Give suitable example. — **4M**

'this' pointer:

1. C++ uses a unique keyword called 'this' to represent an object that invokes a member function.
2. This unique pointer is automatically passed to a member function when it is invoked.
3. 'this' is a pointer that always point to the object for which the member function was called.
4. For example, the function call A.max () will set the pointer 'this' to the address of the object A. Next time suppose we call B.max(), the pointer 'this' will store address of object B.

Explanation 2M

Consider the following example: — ***Example 2M***

```
#include<conio.h>
#include<iostream>
class sample
{
        int a;
        public:
        void setdata(int x)
        {
                this ->a=x;
        }
        void putdata()
        {
                cout<<this ->a;
        }
};
void main()
```

MODEL ANSWER

SUMMER - 2017 EXAMINATION

Subject: Object Oriented Programming **Subject Code:** 17432

4.		**Attempt any FOUR of the flowing:**	**16**
	(a)	**Write a program to implement single inheritance. Declare base class 'Employee' with emp_no and emp_name. Declare derived class 'Fitness' with height and weight. Accept and display data for one employee.**	**4M**

	Ans.	#include<iostream.h>
		#include<conio.h>
		class employee
		{
		protected:
		int emp_no;
		char emp_name[25];
		void getdata()
		{
		cout<<"\n Enter employee no. ";
		cin>>emp_no;
		cout<<"\n Enter emplyee name";
		cin>>emp_name;
		}
		void display()
		{
		cout<<"\n Employee no. is :"<<emp_no;
		cout<<"\n Employee name is:"<<emp_name;
		}
		};
		class fitness:public employee
		{
		protected:
		float height,weight;
		public:
		void getdata()
		{
		employee::getdata();
		cout<<"\n Enter height:";
		cin>>height;
		cout<<"\n Enter weight:";
		cin>>weight;
		}
		void display()
			Correct logic 2M
			
			Correct syntax 2M

```
                void accept()
                {
                        cout<<"enter code of staff:"<<endl;
                        cin>>code;
                }
                void dis()
                {
                        cout<<"code="<<code<<endl;
                }
};
class teacher : public staff
{
        protected:
                char subject[10];
        public:
        void acc1()
        {
                cout<<"enter subject:"<<endl;
                cin>>subject;
        }
        void dis1()
        {
                cout<<"subject="<<subject<<endl;
        }
};
class officer :public staff
{
        protected:
                char grade[5];
        public:
                void acc2()
                {
                        cout<<"Enter Grade:"<<endl;
                        cin>>grade;
                }
                void dis2()
                {
                        cout<<"grade="<<grade<<endl;
                }
```

logic 2M

Correct syntax 2M