0197

National University of Computer and Emerging Sciences
Lahore Campus

Object Oriented Programming (CS1004)

Final Exam

Date: May 24, 2024

Course Instructor(s)

Mr. Aamir Rahim,Ms. Anosha Khan
Ms. Arooj Khalil,Ms. Samin Iftikhar
Mr. Uzair Naqvi,Mr. Waqas Manzoor

Total Time (Hrs):	**3**
Total Marks:	**90**
Total Questions:	**4**

[redacted]
Roll No

[redacted]
Section

[redacted]
Student Signature

IMPORTANT INSTRUCTIONS: Do not use pencil or red ink to answer the questions. In case of confusion or ambiguity make a reasonable assumption, questions during exam are not allowed. **Only one solution will be marked against one question, carefully attempt questions on <u>answer sheet</u>.**

You may freely use built-in string functionality such as strlen, strcpy, strcmp etc. where required. Do not re-write these functions.

Do not write below this line.

Attempt all the questions.

CLO # 4: Apply good programming practices

Q1: [6x5 = 30 marks]

Part (I) Write output of the program given below. If program crashes, clearly mention that. (There is no syntax error in this code.)

```
#include <iostream>
using namespace std;
void mightGoWrong() {
    bool error = true;
    if (error) {
        cout << "In mightGoWrong " << endl;
        throw bad_alloc();
    }
}
void doSomething() {
    try {
        cout << "In doSomething " << endl;
        mightGoWrong();
    }
    catch (bad_alloc e) {
        cout<< "Caught bad_alloc in 
doSomething: " << e.what() << endl;
        throw new runtime_error("Runtime 
error");
    }
}
int main() {
    try {
        cout << "In Main() " << endl;
        doSomething();
    }
    catch (exception e) {
        cout<< "Caught rethrown exception in 
main: " << e.what() << std::endl;
    }
    catch (...) {
        cout << "Caught rethrown ... in main: 
" <<endl;
    }
    return 0;
}
```

Part (II) Write output of the program given below. If program crashes, clearly mention that. (There is no syntax error in this code.)

```
#include<iostream>
using namespace std;
void D()
{
    cout << "Start D\n";
    cout << "D throwing int exception\n";
    throw - 1;
    cout << "End D\n";
}
void C()
{
    cout << "Start C\n";
    D();
    cout << "End C\n";
}
void B()
{
    cout << "Start B\n";
    try
    {
        C();
    }
    catch (double)
    {
        cout << "B caught double exception\n";
    }
    catch (...)
    {
        cout << "B caught default
exception\n";
    }
    try
    {
        throw -1;
    }
    catch (int)
    {
        cout << "B caught int exception\n";
    }
    cout << "End B\n";
}
void A()
{
    cout << "Start A\n";
    try
    {
        B();
    }
    catch (int)
    {
        cout << "A caught int exception\n";
    }
    cout << "End A\n";
}
int main()
{
    cout << "Start main\n";
    try
    {
        A();
    }
    catch (int)
    {
        cout << "main caught int exception\n";
    }
    cout << "End main\n";
    return 0;
}
```

Part(III) Write output of the program given below. (There is no syntax error in this code.)

```
#include<iostream>
using namespace std;
class A {
public:
    int i;
    A(int ii) : i(ii) {
        cout << "Calling A(int ii)" << endl;
    }
    void show() {
        cout <<"A i=" << i << endl;
    }
};
class B {
    int x;
    A obj;
public:
    B(int xx) : x(xx), obj(xx + 5) {
        cout << "B constructor" << endl;
    }
    B(B& o):obj(o.x) {
        cout << "Copy constructor B" << endl;
        x = o.x + 5;
    }
    void show() {
        obj.show();
    }
};
int main()
{
    B b(10);
    B b1(b);
    b1.show();
    return 0;
}
```

rt (IV) Write output of the program given below. (There is no syntax error in this code.)

```
include <iostream>
#include <string>
using namespace std;
class A{
        float d;
   public:
        virtual void func(){
                cout<<"1 2 3 \n"; }
     void func2(){
                cout<<"4 5 6\n"; }
};
class B: public A{
public:
        void func(){
                cout<<"A B C \n";
        }
        void func2(){
                cout<<"X Y Z\n";
        }
};
```

```
void SomeFunction(A& obj)
{
        obj.func2();
}
void AnotherFunction(A& obj)
{
        obj.func();
}
int main()
{
        B b;
        b.func();
        SomeFunction(b);
        AnotherFunction(b);
        A a_obj2 = b;
        a_obj2.func();
        return 0;
}
```

Part (V)

You are given a main function and its required output. **Best using the practices studied in the class, write missing functionality such that this program runs successfully and gives required output.** You are NOT ALLOWED to change main program and there is no syntax error in this code. *No credit will be given for bad solution.*

Main Program

```
#include<iostream>
using namespace std;
int main() {
    int intArr[] = { 5, 2, 9, 1, 6 };
    int intSize = sizeof(intArr) / sizeof(int);
    MySort(intArr, intSize);
    cout << "Sorted integer array:\n";
    for (int i = 0; i < intSize; ++i) {
        cout << intArr[i] <<endl;
    }

      float floatArr[] = { 5.2, 9.7, 2.6, 6.5, 1.9 };
    int floatSize = sizeof(floatArr) / sizeof(float);
    MySort(floatArr, floatSize);
    cout << "Sorted floats array:\n";
    for (int i = 0; i < floatSize; ++i) {
        cout << floatArr[i] <<endl;
    }

    const char* strArr[] = { "banana", "apple", "orange",
"grape", "kiwi" };
    int strSize = sizeof(strArr) / sizeof(char*);
    MySort(strArr, strSize);
    cout << "Sorted string array:\n";
    for (int i = 0; i < strSize; ++i) {
        cout << strArr[i] <<endl;
    }
    return 0;
}
```

Required Output

```
Sorted integer array:
1
2
5
6
9
Sorted floats array:
1.9
2.6
5.2
6.5
9.7
Sorted string array:
apple
banana
grape
kiwi
orange
Press any key to continue . . .
```

CLO # 3: Model an algorithmic solution for a given problem using OOP

Q2: [15 marks]

We are dealing with a videos application that maintains a list of videos and offers different features. One of its required features is to suggest videos to users according to their preferences. For this, it maintains a list of tags (*TagsList*) against each video indicating different categories the video belongs to.

Examples of *TagsList* of few videos are given below:
TagsList for a video: *{"EidCollection", "Summer2024", "EidUlAzha", "Lawn2024"}*
TagsList for another video: *{"Politics", "ECP", "Election2024", "Pakistan"}*
TagsList for another video: *{"Food", "Travel", "Tourism", "NorthernAreas", "Pakistan"}*

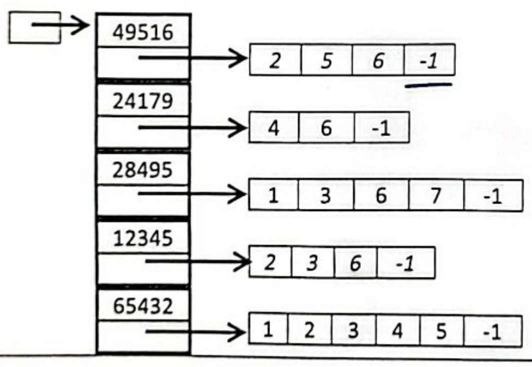
User preferences are also recorded as list of tags, for example:
User1 Preferences: *{"Politics", "CurrentAffairs", "Pakistan"}*
User2 Preferences: *{"Gardening", "Travel", "Food"}*

Application will be suggesting a video to a user if any two tags from user's preferences are found in the TagsList of that video.
For simplicity, the system assigns Tag_ID (int) to each Tag as shown in the table below, and instead of saving list of strings for *TagsList*, it saves list of integers (their corresponding IDs).

```
//Partial Definitions and already
//given functions of classes Video
//and VideoSystem
class Video{
        int* TagsList;
        int ID;
public:
        int* GetTagsList();
        int GetID();
        //Other functions...
};
class VideoSystem{
        Video* AllVideos;
        int TotalVideos;
public:
        //Other Functions...
};
```

Tag_ID	1	2	3	4	5	6	7	and
TagDescription	"Gardening"	"Politics"	"Travel"	"Sports"	"CurrentAffairs"	"Pakistan"	"Food"	so on

Partial definitions of classes Video and VideoSystem are given. Suppose *TotalVideos = 5*; rest of the data of VideoSystem is explained below:

Class Member	Current Value	Explanation
TotalVideos (int)	5	Total number of Videos in the system
AllVideos (Video)*	 49516 2 5 6 -1 24179 4 6 -1 28495 1 3 6 7 -1 12345 2 3 6 -1 65432 1 2 3 4 5 -1	Array of Videos, where - *Video1(ID=49516)* has *Tags:{"Politics","CurrentAffairs","Pakistan"}* - *Video2(ID=24179)* has *Tags:{"Sports","Pakistan"}* - *Video3(ID=28495)* has *Tags:{"Gardening","Travel", "Pakistan","Food"}* and so on.

Your task is to write a function
"int* VideoSystem::GetSuggestedVideosIDsByUserPreferences(int* userPref)"
that takes a user's preferences (int*) and returns IDs of all the videos (int*) suggested by the system.

Sample Run: Output of this function for "**User1 Preferences**" and "**User2 Preferences**" (mentioned above) would be: ***{49516, 12345, 65432,-1}*** and ***{28495, 65432, -1}*** respectively. *Efficient utilization of memory is NOT required.*

Safely assume that you are already given fully functional classes Video and VideoSystem and they are already successfully providing different other features and functions (constructors, destructors etc.). You may freely use already given functions, if required. **You are NOT required to re-write already given functions.**

LO # 1: Demonstrate the basic concepts of OOP

Q3: [8+7 = 15 marks]

Part (A): You are given a driver program (main) along with partial implementation of required classes. Expected output of the driver program is also given. **Your task is to write additional function(s) required (if any) such that the driver successfully gives expected output.** Do not re-write already given code. Clearly mention all the changes required. You are also not allowed to change main program. (Note: There is no syntax error in the given code.)

Partial Code:

```
#include<string>
#include<iostream>
using namespace std;

class Date{
        int Day, Month, Year;
public:
        Date(int d=1, int m=1, int y=1960): Day(d), Month(m), Year(y)
        {
        }
        void Print(){cout<<Day<<"-"<<Month<<"-"<<Year;}
};
class Employee{
        string Name;
        Date DOB;
        Date JoiningDate;
public:
        Employee(string name="", int d1=1, int m1=1, int y1=1960, int d2=1, int m2=1, int
y2=1960): Name(name), DOB(d1,m1,y1), JoiningDate(d2, m2, y2)
        {
        }
        void Print(){
                cout<<"Name: "<<Name<<endl;
                cout<<"DOB: "; DOB.Print(); cout<<endl;
                cout<<"Joining Date: "; JoiningDate.Print(); cout<<endl;
        }
};
void main()
{
        Employee emp("Abc Xyz", 29, 12, 2000, 10, 6, 2020);
        emp.Print();

        emp["DOB"][0] = 15;
        emp["JoiningDate"][2] = 2022;

        cout<<"--------------------"<<endl;
        emp.Print();
}
```

Expected Output:

```
Name: Abc Xyz
DOB: 29-12-2000
Joining Date: 10-6-2020
--------------------
Name: Abc Xyz
DOB: 15-12-2000
Joining Date: 10-6-2022
Press any key to continue . . .
```

Part (B): A piece of code and its output is given below where a Person has a Date of Birth (DOB). Recall the properties of Composition that we studied in class. **Your task is to modify the given code to implement properties of Composition between Person and DOB using Date* DOB.** *You are NOT ALLOWED to have Date object (Date DOB) in Person class*. If required, you may add as many functions or code as you want to fulfill this task. In your answer, you are required to write only modified/newly added functions/code (if any). You are not required to re-write the function(s) which will not change. Properly write the functions/code (if any) to get credit. (Note: There is no syntax error in the given code.) Output should remain the same after modification(s).

```
#include<string>
#include<iostream>
using namespace std;

class Date{
	int Day, Month, Year;
public:
	Date(int d=1, int m=1, int y=1960): Day(d), Month(m), Year(y)
	{
	}
	void Print(){cout<<Day<<"-"<<Month<<"-"<<Year;}
};
class Person{
	string Name;
	Date* DOB; //Date of Birth
public:
	Person(string name)
	{
		Name = name;
		DOB = 0;
	}
	void Print(){
		cout<<"Name: "<<Name<<endl;
		if (DOB)
			cout<<"DOB: "; DOB->Print(); cout<<endl;
	}
	void SetDOB(Date* dptr){DOB = dptr;}
};
int main() {
	Date dob(18,5,2005);
	Person p1("Dummy Employee");
	p1.SetDOB(&dob);

	p1.Print();
	return 0;
}
```

```
Output:
Name: Dummy Employee
DOB: 18-5-2005
Press any key to continue . . .
```

National University of Computer and Emerging Sciences
Lahore Campus

CLO # 2: Apply OOP concepts (Encapsulation, Inheritance, Polymorphism, Abstraction) to computing problems for the related program

Q4: [30 marks]

We have to make an application that manages Contacts. A **Contact** has a *name (string)* and it may have *ContactDetails*. A **ContactDetail** can be a PhoneNo, Email, Address, Website etc. We are only dealing with Phone and Address due to shortage of time. Every **ContactDetail** has a *Type (integer)*. Detail about the type is given in the tables below. You just have to save an integer for type and don't worry about its corresponding value. A **PhoneNo** has a *Number (string)*. An **Address** has a *Street (string), City (string), State (string), PostCode (string)* and *Country (string)*.

Your task is to provide fully functional classes (using concepts of OOP) required to run the given main program successfully. ***You are required to best use the practices studied in class.*** Make sure that your program doesn't leak any memory and it doesn't crash. You may use string OR c-string for data storage where required. Also you do not need to re-write same code with different variables again and again.
For example, instead of writing: "Function1Call(var1); Function1Call(var2); Function1Call(var3);"
just write: "Function1Call(var1); //Same code as above for var2, var3"

Detail about Type of Address:

Type(int)	1	2	3
Values against each Type	"Home"	"Work"	"Other"

Detail about Type of Phone:

Type(int)	1	2	3	4
Values against each Type	"Mobile"	"Home"	"Work fax"	"Other"

Safely assume that following global functions are already given and you can freely use them where required. **You are not required to re-write these functions.**

```
char* GetAddressType(int type)
{
        if (type == 1) return
"Home";
        else if(type == 2)
return "Work";
        else return "Other";
}
```

```
char* GetPhoneType(int type)
{
        if (type == 1) return
"Mobile";
        else if(type == 2)
return "Home";
        else if(type == 3)
return "Work fax";
        else return "Other";
}
```

```
char* StringDeepCopy(char*
src)
{
        char* dest = new
char[strlen(src)+1];
        strcpy(dest,src);
        return dest;
}
```

Main Program: (Note: There is no syntax error in the given code and you are not allowed to change main.)

```
void main()
{
        Contact contact1("Ali Hamza");    //Creates a contact with Name = Ali Hamza and no
ContactDetails. Name passed to this function will always be valid.

        contact1.Print();
        cout<<"--------------------------------"<<endl;

        Contact contact2("Usman Khalid"); //Creates another contact with Name = Usman Khalid
and no ContactDetails

        contact2.Print();
        cout<<"--------------------------------"<<endl;

        //Following lines are adding contact details in contact 1 and contact 2:
        contact1.AddContactDetail(new PhoneNo(1,"0300-1234567"));      //Adds a ContactDetail, a
PhoneNo with Type = 1(means Mobile) and Number = 0300-1234567. Safely assume that this
function will always get valid values for all the data.
```

```
	contact1.Print();
	cout<<"--------------------------------"<<endl;

	contact1.AddContactDetail(new PhoneNo(3,"042-111-128-128"));  //Adds another
ContactDetail, a PhoneNo with Type = 3(means Work Fax) and Number = 042-111-128-128
	contact1.AddContactDetail(new Address(2, "852-B Milaad St, Block B Faisal Town",
"Lahore", "Punjab", "54770", "Pakistan"));	//Adds another ContactDetail, an Address with
Type = 2(Work), Rest of the values are for Street, City, State, PostCode and Country
respectively.
	//Safely assume that this function will always get valid values for all the data.
	contact1.AddContactDetail(new Address(1, "853-B Faisal Town", "Lahore", "Punjab",
"54770", "Pakistan")); //Adds another Address here Type = 1(Home)

	contact2.AddContactDetail(new Address(3, "123-B Xyz Town", "Gujranwala", "Punjab",
"12345", "Pakistan")); //Adds an Address in contact 2 here Type = 3(Other)

	contact1.Print();
	cout<<"--------------------------------"<<endl;
	contact2.Print();
	cout<<"--------------------------------"<<endl;
}
```

Required Output:

```
[Name] Ali Hamza
--------------------------------
[Name] Usman Khalid
--------------------------------
[Name] Ali Hamza
[Mobile] 0300-1234567
--------------------------------
[Name] Ali Hamza
[Mobile] 0300-1234567
[Work fax] 042-111-128-128
[Work] 852-B Milaad St, Block B Faisal Town
Lahore,Punjab 54770
Pakistan
[Home] 853-B Faisal Town
Lahore,Punjab 54770
Pakistan
--------------------------------
[Name] Usman Khalid
[Other] 123-B Xyz Town
Gujranwala,Punjab 12345
Pakistan
--------------------------------
Press any key to continue . . .
```