

Student name: _____ Student ID: _____

Cairo University
Faculty of Computers and Artificial
Intelligence
Subject: Object Oriented Programming
Subject Code: CS213
Examiner(s): Dr. Hanaa Bayoumi
Dr. Cherry Ahmed



Mid-term exam
Date: 20/11/2023
Semester: 1st
Duration: **1 hour**

/40

/20

Question 1: Choose one correct answer: [/10 marks] (2 pts each)

1. If a function is friend of a class, which one of the following is **wrong**?
 - a) **Friend functions are members of a class**
 - b) A function can only be declared as a friend by the class itself
 - c) It can have access to all variables of a class, even private ones
 - d) The function can be a member of another class or a stand-alone function
2. Which member function of class cannot modify its objects attributes?
 - a) Inline member functions
 - b) Private member functions
 - c) Static member functions
 - d) **Constant member functions**
3. The method that moves the write pointer of an fstream file is:
 - a) seekg()
 - b) **seekp()**
 - c) tellg()
 - d) tellp()
4. Which of the following statements call the method header → **A operator++(int) {...}**
 - a) **A a_obj; a_obj++;**
 - b) **A a_obj; a_obj++(10);**
 - c) **A a_obj; ++a_obj;**
 - d) None of the above
5. For the class **Circle**, a member function **F** is defined as follows:
[return type] F() { return this; }
What is the return type of this function?
 - a) Circle
 - b) **Circle***
 - c) Circle&
 - d) None of the previous

Question 2: Answer the following: [/ 14 marks]

1. This code has multiple **syntax errors**: (6)

(A) Identify each by mentioning the line number and the cause of the error

(B) Fix them by correcting/editing the given code.

Line 1:	class Person	Line 11:	int main()
Line 2:	{	Line 12:	{
Line 3:	string name;	Line 13:	Person people[10];
Line 4:	int grades[5];	Line 14:	Person p1("John");
Line 5:	public:	Line 15:	Person * p2 = &p1;
Line 6:	Person(string n):name(n){}	Line 16:	cout<<p2.getName();
Line 7:	void setName(string n){name=n;}	Line 17:	p1[0] = 95;
Line 8:	string getName(){ return name;}	Line 18:	return 0;
Line 9:	int operator[] (const int ind)		}
Line 10:	{ return grades[ind];}		
	};		

Line 13: The array declaration requires a default constructor which is not available.

Fix: Add a default constructor to Person: Person(){ }

Line 16: p2 is a pointer, so the dot notation can't be used

Fix: Use arrow: cout<<p2->getName();

Line 17: Trying to set an element in the grades array while the overloaded operator returns an int.

Fix: the overloaded operator [] should return int&

class Test	int main()
{	{
public:	Test t1,t2;
static int cnt;	Test *t3, *t4;
Test(){ cnt++; }	t4 = new Test(5);
Test(int n){ cnt+=n; }	cout<<t1.cnt<<" ";
~Test(){ cnt--; }	delete t4;
};	cout<<Test::cnt;
int Test::cnt = 10;	return 0;
	}

2. What is the output of the following code? (4)

OUTPUT: 17 16 (2 pts each)

3. What is the output of the following code? (4)

OUTPUT: 11 11 (2 pts each)

```
class Num
{
    int* p;
public:
    Num () { p = new int; }
    Num (const Num & obj)
    {
        p = new int;
        set(*(obj.p));
    }
    void set(int n) { *p = n; }
    int get() { return *p; }
    ~Num () { delete p; }
};
```

```
int main()
{
    Num t1,t2;
    t1.set(7);
    t2 = t1;
    t2.set(11);
    cout<<t1.get()<<" ";
    Num t3 = t2;
    t3.set(9);
    cout<<t2.get();
    return 0;
}
```

Question 3: Answer the following: [/ 16 marks]

Implement a class **Date** that represents a date as a month and a year.
The class has the following members:

- The private members month & year, where month ranges from 1 to 12, and year is a positive number. (1)
- A private function **bool validate(int mo,int yr)** which checks for the validity of the given parameters month and year and returns true if they have valid values and false otherwise. (3)
- A parameterized constructor that takes a month and a year as parameters. The constructor sets the members only if the given values are valid (use the function validate), otherwise an error message is displayed. (3)
- Overload the operator+=(int) such that number of months is given to be added to the month. Make sure that if month becomes greater than 12 after addition, add to the year and adjust month. Return the updated Date object. (3)
- Overload the operator>= which returns true if the first date is greater than (comes after) or equal to the second and false otherwise. For ex., the date 10/2020 is greater than 8/2020. (3)

Implement a **friend function** void toFile(ofstream& out, const Date& d) which takes a file output file stream, and a date object. The function writes the month and year of the parameter Date object on a separate line as "[month] / [year]". (3)

The file **dates.txt** is shown after the execution of the following main:

```
int main()
{
    Date d1(4,2010), d2(12, 2022);
    ofstream ofile("dates.txt",ios::app);

    toFile(ofile,d1);
    toFile(ofile,d2);
    d1+= 10;
    toFile(ofile,d1);
    d2 += 1;
```

```
toFile(ofile,d2);
ofile.close();
//This outputs D1 < D2
if(d1 >= d2)
    cout<<"D1 >= D2";
else
    cout<<"D1 < D2";
return 0;
}
```

```
4/2010
12/2022
2/2011
1/2023
```

dates.txt

```

class Date
{
    int month, year; (1 pt)
    bool validate( int m, int y) (0.5 pt defined in private section)
    {
        if(m<1 || m>12 || y<=0) return false; (1.5 pt)
        else return true; (1 pt)
    }
public:
    Date(int m, int y) (0.5 pt)
    {
        if(validate(m,y)) (1 pt)
        {
            month = m; (0.5 pt)
            year = y; (0.5 pt)
        }
        else
            cout<<"Invalid month or year"; (0.5 pt)
    }
    bool operator>=(const Date& right) (1 pt)
    {
        if((month == right.month) && (year == right.year)) return true; 0.5
        else if(month > right.month && year == right.year) return true; 0.5
        else if(year > right.year) return true; 0.5
        else return false; 0.5
    }
    Date operator+=(int months) (0.5 pt)
    {
        month += months; (0.5 pt)
        if(month>12) (0.5 pt)
        {
            year += month/12; (0.5 pt)
            month %= 12; (0.5 pt)
        }
        return *this; (0.5 pt)
    }
    friend void toFile(ofstream&,const Date&); (1 pt)
};
void toFile(ofstream& ofile,const Date& d) (1 pt)
{
    ofile<<d.month<<"/"<<d.year<<endl; (1 pt)
}

```