

I-

Write the Interface, implementation and driver for the following:  
(59 points)

Create a *Time* Class. Each object of this class will represent a specific time and name of the day, storing the hours, minutes, and seconds as integers and day as `char*`. Include appropriate constructor(s), destructor, access methods (set and/or get), a method `advance(int h, int m, int s)` to advance the current time of an existing object, a method `reset()` to reset the current time of an existing object to 1:00:00 AM, a `print()` function that prints the day and time, and a function `getcount()` that returns the number of objects created.

Create a non-member function called `fillarray()` that fills up the array with the corresponding number of objects.

Create a non-member function called `sortarray()` that sorts the array of objects by name of the day.

**You must perform the following tasks in the driver:**

- 1- Create a dynamic array of n objects and call it `schedule`, n must be input from the keyboard. int n; Time \* T = new Time [n];
- 2- Call the function `fillarray()`
- 3- Call the function `sortarray()`
- 4- Print array objects without call a function.
- 5- Create automatic object with ("wed", 10, 10, 10) as initial values and call the function `print`. Time T ("wed", 10, 10, 10);
- 6- Create a dynamic object with the default initial values.
- 7- Set the day to `Sunday` and hour of the dynamic object to `12` and call the `print` function
- 8- Print the number of objects created.
- 9- Make sure that dynamic objects are properly de-allocated.

II- Circle the correct answer (2 point each)

1- Member function definitions

- (a) always require the binary scope operator (::).
- (b) only require the binary scope operator when being defined outside of the scope of their class.
- (c) can use the binary scope operator anywhere, but become public functions.
- (d) must use the binary scope operator in their function prototype

2- A class may contain multiple constructors if

- (a) they have different names.
- (b) they have different argument lists.
- (c) they have the same argument list.
- (d) they have different return types.

3- Every object of the same class

- (a) gets a copy of every function and member variable
- (b) gets a copy of every member variable.
- (c) gets a copy of every function.
- (d) shares pointers to all member variable and fun

4- The proper format for a struct is

(a) struct Time

int hour

int minute;

(b) struct Time {

int hour,

int minute,

}

(c) struct Time {

int hour;

int minute;

}

(d) struct Time {

int hour,

int minute;

};

5- Changing the value of private member variables is

(a) only possible with private member functions.

(b) possible using public functions and references.

(c) impossible.

(d) only possible if the private variables are not declared inside the class.

6- Variables defined inside a member function of a class have

(a) file scope.

- (b) class scope.
- (c) function scope.
- (d) class or function scope, depending on whether the binary scope resolution operator (::) is used.

7- timePtr is a pointer to object timeObject with data member hour. Which of the following is not equivalent to hour?

- (a) timeObject.hour
- (b) timePtr.hour
- (c) timePtr->hour
- (d) (\*timePtr).hour

8- Member access specifiers (public and private) can appear

- (a) in any order and multiple times.
- (b) in any order (public first or private first) but not multiple times.
- (c) in any order and multiple times, if they have brackets separating each type.
- (d) multiple times, but all specifiers of the same type must be grouped together.

9- A default constructor

- (a) is a constructor with all default arguments
- (b) is the constructor generated by the compiler when one is not provided by the programmer
- (c) does not perform any initialization
- (d) both (b) and (c)

10- By default, structures are passed

- (a) call-by-value
- (b) call-by-reference
- (c) one member at a time
- (d) structures cannot be passed between functions

III- Suppose your program contains the following class definition. (21 points)

```
class Automobile
{
public:
void setPrice(double newprice);
void setprofit(double newProfit);
double getPrice();

private:

double price;
double profit;
double getProfit();

};
```

and suppose the *main* function of your program contains the following declaration and that the program somehow sets the values of all the member variables to some values:

```
Automobile hyundai, jaguar;
```

Which of the following statements are then allowed in the main function of your program?

```
hyundai.price = 4999.99;
```

```
jaguar.setPrice(30000.97);
```

```
double aprice, aprofit;
```

```
aprice=jaguar.getPrice();
```

```
aprofit=jaguar.getProfit();
```

```
aprofit=hyundai.getProfit();
```

```
hyundai=jaguar;
```