

Practice Midterm sample solution

COP 3331, Fall 2018

“I pledge my honor that I have not cheated, and I will not cheat on this exam. I understand that if I have been found to violate this pledge, I will receive a score of 0 for this exam, and I may receive an FF for the course.”

Signed: _____

1. (5 points) Which of the following lines of code would you use when prompting a user for their first and last name?

☐ `cin >> fullname;`
☐ `cin << fullname;`
☐ `fullname = cin.getline()`
☒ `getline(cin, fullname);`

2. (5 points) How many spaces are printed out by the following statement:

```
cout < fixed << setprecision(2) << setw(10) << 3.14159265;
```

☐ 0
☐ 2
☐ 4
☒ 6
☐ 8
☐ 10

3. (5 points) Which of the following would be the *best* prototype for a `toString` function that converts a **vector** of **ints** into a C-style string?

☐ `char* toString(vector<int> vec)`
☒ `char* toString(const vector<int>& vec)`
☐ `char* toString(vector<int>&& vec)`
☐ `char* toString(vector<int> vec) const`
☐ `char* toString(vector<int>& vec) const`

4. (5 points) Which of the following lines of code will result in a compiler or runtime error?

☐ `array<int, 5> arr {1, 2, 3, 4, 5};`
☐ `for (int i = 0; i < arr.size(); i++)`
☐ `cout << arr[i] << ' ';`
☒ **None of the above will cause an error**

5. Answer the following about developing a `Student` class to represent students at USF.

- (a) (5 points) Implement a `UnumToInt` function that takes in a string containing a U number (e.g., "U12345678") and returns an `int` containing the number part of the U number (12345678).

Solution:

```
int UnumToInt(string str)
{
    istringstream iss(str);
    char u;
    int num;
    iss >> u; //get rid of 'U'
    iss >> num;
    return num;
}
```

- (b) (5 points) Write a move constructor for the `Student` class. The data members of the class are `int unum`, `char* fname`, and `char* lname`.

Solution:

```
Student::Student(Student&& move)
{
    unum = move.unum;
    fname = move.fname;
    move.fname = nullptr;
    lname = move.lname;
    move.lname = nullptr;
}
```

6. (a) (25 points) Write a class definition (as in, what would go in your header file) for a `Point2D` class that represents a point in 2D space. Your class should have 4 member functions:
1. A default constructor
 2. A two-value constructor that accepts two `double` values
 3. A `translate` function that accepts a `Point2D` reference and returns a `Point2D` object. This member function is an accessor.
 4. A `rotate` function that accepts a `double` and returns nothing. This member function is a mutator.

In addition, your class should grant access to its data members to a print function that can be used with `cout`.

Solution:

```
class Point2D
{
    friend ostream& operator<<(ostream&, const Point2D&); //2nd arg may vary
private:
    double x;
    double y;
public:
    Point2D();
    Point2D(double, double);
    Point2D translate(Point2D) const;
```

```
void rotate(double);
};
```

- (b) (5 points) Would it be acceptable or unacceptable to use the default copy constructor for `Point2D`? Justify your answer.

Solution: Acceptable, since a `Point2D` class almost certainly doesn't have any pointer data members

- (c) (10 points) Implement the print function for `Point2D`. This function should be usable with `cout` (i.e., `cout << point << end;` should work). It should output the `Point2D` as an ordered pair; e.g., (1, 0) or (2.718, 3.1415).

Solution:

```
ostream& operator<<(ostream& out, const Point2D& p)
{
    return out << '(' << p.x << ", " << p.y << ')';
}
//second arg could be Point2D or reference instead
```

- (d) (10 points) The `rotate` function should alter the given `Point2D` by rotating it a given number of radians counterclockwise about the origin. Describe at least 3 meaningful test cases for `rotate`, as well as the expected outcome when rotating the point (1, 0) by these values. You may use standard mathematical notation when describing your test cases (not necessarily how they would appear in code).

Solution: Many possible answers

Test case	Expected result for (1, 0)
0	(1, 0)
2π	(1, 0)
$\frac{\pi}{4}$ (or some other value between 0 and 2π)	$(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$
$-\frac{\pi}{2}$ (or some other negative value)	(0, -1)
3π (or some other value $> 2\pi$)	(-1, 0)

7. (20 points) You are working for a new social media start-up that is developing a platform to compete with Facebook. Draw a UML diagram for a class that represents a user of your platform. At a minimum, your class should have three data members, two constructors, two accessors, and two mutators.

User
<p>Many possible data members, e.g.:</p> <ul style="list-style-type: none"> - name : string - password : string - address : Address - dob : Date - id : user_id_t - picture : photo_id_t - friend : vector<user_id_t> - post : vector<post_id_t> - pic : vector<photo_id_t>
<pre> + User(email : string, pass : string) + User(copy : const User&) ... + changeName(newname : string) : void + addFriend(newfriend : user_id_t) : void + addPicture(p : const Photo&) : void + makePost(p : const Post&) : void + setProfilePic(p : photo_id_t) : void ... + getName() const : string + verifyPassword(pw : string) const : bool + sendEmail(contents : string) const : void + getID() const : user_id_t + getAge() const : int + getHomepage() const : HTML ... </pre>

Solution: