Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Software Engineering II (CS 562)

Exam 1

Part I – True / False, circle T for true or F for false. Statements are intended to have one answer. Evaluate truth based upon the material covered in this class (20 points, 2 points per question).

1. T F Agile development may include daily meetings, acceptance testing, iterations that last from one week to one month, and release cycles.
2. T F Extreme Programming (XP) is an agile method that focuses on teamwork, especially pair programming, and uses code as one of the main means of communication.
3. T F Scrum includes three core development roles – product owner, scrum master, and developers.
4. T F An equivalence class is a set of data that represents a single test condition.
5. T F Livelock is a condition in which processes needing to use the same resource are all halted because they are all waiting to use that resource. These processes consume no CPU cycles while waiting.
6. T F Deadlock may always be avoided by changing the order in which resources are allocated to processes.
7. T F The pipe and filter pattern models a stream of data passed through a group of processes referred to as a pipeline.
8. T F Model driven development includes creating domain models, ER diagrams, and use cases; refining component interactions, component interfaces, data distribution and functionality, and reusable components; reviewing use cases and architecture feasibility; and finalizing interfaces and class/interaction diagrams.
9. T F Design principles include but are not limited to increasing cohesion, decreasing coupling, and anticipating obsolescence.
10. T F Software architecture is the process of designing system's global organization, dividing it into subsystems, deciding how subsystems will interact, and determining the interfaces.

Part II – Short and long answer questions (80 points).

1. Define the parts of the Model View Controller architecture and provide an example of each part (10 points).
2. Below are examples of five types of errors: null condition not handled, off-by-one, improper precision, infinite loop, and improper floating point comparison. For each example list the type of error, and explain why it is of that type. Assume one error per sub-problem, and assume any comments provided are correct (20 points, 4 points each).

**A)** int n = 10;

double d = 0.0;

for(int i = 0; i < n; i++)

 d += 2.0;

if(d == 20.0)

 System.out.println("d is 20.0");

**B)** //Print all the elements in an array.

int [] arr = { 10, 9, 8, 7, 6 };

int n = arr.length;

for(int i = 1; i < n; i++)

 System.out.println(arr[i]);

**C)** //Do processing if "yes" or "no" is provided as input.

Scanner scan = new Scanner(System.in);

while(true){

 int input = 0;

 System.out.println("Enter yes or no.");

 String line = scan.nextLine().trim();

 if(line.equals("yes")) input = 1;

 else if(line.equals("no")) input = 2;

 else input = 3;

 switch(input) {

 case 1: case 2: case 3:

 System.out.println("Invalid input, please enter yes or no.");

 default:

 break;

 }

}

**D)** //Insert an element at the front of a linked list.

void insert(LLElement e) {

 if(head == null)

 head = e;

 else {

 e.next = head;

 head = e;

 }

}

**E)** //The following code moves a robot arm designed for surgery

float [] move(float phi, float psi, float theta, float [] vec)

{

 float [] retVec = new float[3];

 float cosT = Math.cos(theta), sinT = Math.sin(theta);

 float cosPhi = Math.cos(phi), sinPhi = Math.sin(phi);

 float cosPsi = Math.cos(psi), sinPsi = Math.sin(psi);

 retVec[0] = cosT\*cosPsi\*vec[0] + (-cosPhi\*sinPsi + sinPhi\*sinT) \* vec[1] + (sinPhi\*sinPsi + cosPhi\*sinT\*cosPsi) \* vec[2]

 retVec[1] = cosT\*sinPsi\* vec[0] + (cosPhi\*cosPsi + sinPhi\*sinT\*sinPsi)\* vec[1] + (-sinPhi\*cosPsi + cosPhi\*sinT\*sinPsi) \* vec[2];

 retVec[2] = -sinT\* vec[0] + sinPhi\*cosT\* vec[1] + cosPhi\*cosT\* vec[2];

 return retVec;

}

1. What is the difference between white/glass box and black box testing (10 points)?
2. In your own words and using diagrams describe deadlock, livelock, and how the two are similar and different (10 points).
3. Design by contract is an important part of defensive design. When developing methods or code what are the three important guarantees a programmer must make when implementing a design by contract. Describe each of these (10 points).
4. Complete the following crossword puzzle (10 points).

Word bank: backlog, sprint, user story, unit test, pair programming, cost-benefit analysis, stakeholders, release


1. Describe equivalence classes. What are the equivalence classes and boundary conditions for the following if statement (10 points)?

if( x > 10 && y < 20)

 //Do something.