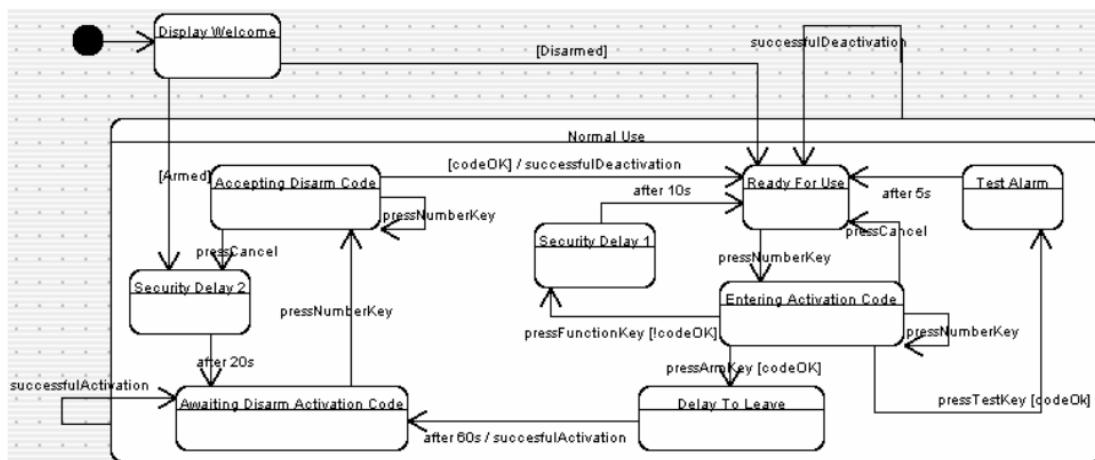


“Instruction to Software Engineering” Final Examination(B)

(闭卷, 120 分钟)

1. Please describe the differences between following items briefly. (15 Points)
 - a) Framework and Pattern;
 - b) Requirement and Specification;
 - c) Construction and Development;
 - d) Verification and Validation;
 - e) Coupling and Cohesion;
2. Provide an example of software project that would be amenable to the incremental process model. Be specific. what an “increment” means in terms of project work and delivery.(10 Points)
3. What work products result from the requirements engineering process?(6 Points)
4. List three golden rules that should be applied when building any user interface.(8 Points)
5. List three areas in which process models may differ from one another. (6 Points)
6. Following is the behavior of the Control Panel user interface for SafeHome system. Please design the GUI interface draft for the Control Panel. (15 Points)



7. The department of public works for a large city has decided to develop a Web-based potholes tracing and repair system (PHTRS 城市道路坑洼跟踪和修复系统). A description follows (16 Points):

Citizens can log onto a website and report the location and severity of potholes. (市民可以登录网站，报告道路坑洼的位置和严重程度。) As potholes are reported they are logged within a “public works department repair system” and are assigned an identifying number , stored by street address, size(on a scale of 1 to 10),location(middle, curb, etc.),district(determined from street address),and repair priority (determined from the size of the pothole). (当报告了道路坑洼时，系统会在“公共工程部门维修系统”内记录信息，分配一个识别号码，记录坑洼的地址、大小(范围从 1 - 10)、位置(路中间、路边等)、地区(根据街道地址确定)和修复优先级(根据坑洼的大小确定)。) Work order data are associated with each pothole and include pothole location and size ,repair crew identifying number, number of people on crew, equipment assigned, hours applied to repair, hole status(work in progress, repaired, temporary repair, not repaired), amount of filler material used, and cost of repair(computed from hours applied, number of people, material and equipment used). (工作订单数据与每个道路坑洼相关联，并包括道路坑洼位置和大小，修理小组识别号码，修缮小组的人数，分配的设备、修复的小时数，修复的状态(正在修复、已经修复、临时修复、没有修复)，使用的填充材料和维修的成本(按照所用小时数、使用的人员数、使用的材料和设备来计算))。 Finally, a damage file is created to hold information about reported damage due to the pothole and includes citizen's name, address, phone number, type of damage, and dollar amount of damage. (最后，将创建一个赔偿文件，以记录由于道路坑洼而造成的损失信息，包括公民的姓名、地址、电话号码、损坏类型和赔偿金额。) PHTRS is an online system; all queries are to be made interactively. (PHTRS 是一个在线系统；所有查询都是交互式进行的。)

- (1) Please create a E-R Diagram or Class Diagram to describe the data for the PHTRS system.
- (2) Please draw a UML Use Case Diagram for the PHTRS system.
- (3) Please develop an Activity Diagram to describe the citizens' reporting an issue of potholes in PHTRS system.
- (4) Please develop a sequence Diagram to describe the citizens' logging in PHTRS system.

Notes: You'll have to make a number of assumptions (假设条件) about the PHTRS system.

8. Given the program logic as follows: (12 Points)

```
int    a, b;
int    x=0;
int    y=0;
if( a>=b ) {
    x = a-b;
} else {
    x = b-a;
}
while ( b<0 and a≠0 ) {
    y += b;
    b++;
}
```

(1) Please draw the Flow Chat for this program. (4 Points)

(2) Please compute the cycle complexity of program. (3 Points)

(3) Please list a set of independent path and design a set of test case for basic path testing.

(5 Points)

9. There is an example of component design that violated both principles of OCP(Open-Closed Principle) and LSP(Liskov Substitution Principle). Please explain why the design violated OCP and LSP . (12 Points)

```
1 struct Point (double x, y);
2
3 struct Shape {
4     enum ShapeType { square, circle } itsType;
5     Shape(ShapeType t) : itsType(t) {}
6 }
7
8 struct Circle: public Shape
9 {
10     Circle(): Shape(circle) {};
11     void Draw() const;
12     Point itsCenter;
13     double itsRadius;
14 }
15
16 struct Square: public Shape
17 {
18     Square(): Shape(square) {};
19     void Draw() const;
20     Point itsTopLeft;
21     double itsSide;
22 }
23
24 void DrawShape(const Shape& s)
25 {
26     if (s.itsType == Shape::square)
27         static_cast<const Square&>(s).Draw();
28     else if (s.itsType == Shape::circle)
29         static_cast<const Circle&>(s).Draw();
30 }
```