Midterm

Note

This is an open-book exam. You may consult any books, papers, or notes, but discussion is strictly forbidden.

Problems

1. Prove the following sequents using Gentzen's System LK. You may take the set view of a sequent to shorten your proofs. (10 %)

   (a) $\vdash (A \to B) \to (\neg B \to \neg A)$

   (b) $\vdash \exists x(P(x) \wedge Q(x)) \to (\exists y P(y) \wedge \exists z Q(z))$

2. The first-order theory for *monoids* contains the following two axioms:

   - $\forall a \forall b \forall c(a \cdot (b \cdot c) = (a \cdot b) \cdot c)$. (Associativity)
   - $\forall a((a \cdot e = a) \wedge (e \cdot a = a))$. (Identity)

   Here e is a constant, called the identity, and $\cdot$ is the binary operation. Let M denote the set of the two axioms. Prove using Gentzen's System LK the sequent $M \vdash \forall e'(\forall a((a \cdot e' = a) \wedge (e' \cdot a = a)) \to e' = e)$, which says that the identity element of a monoid is unique. (Hint: a typical proof in algebra books is the following: assuming e' is an identity, $e' = e' \cdot e = e$.) (15 %)

3. For each of the following informal requirement descriptions, write a formal specification in the form of "$\{pre\}\ y := ?\ \{post\}$" where y is a single variable or a list of variables that may be changed by the program. (10 %)

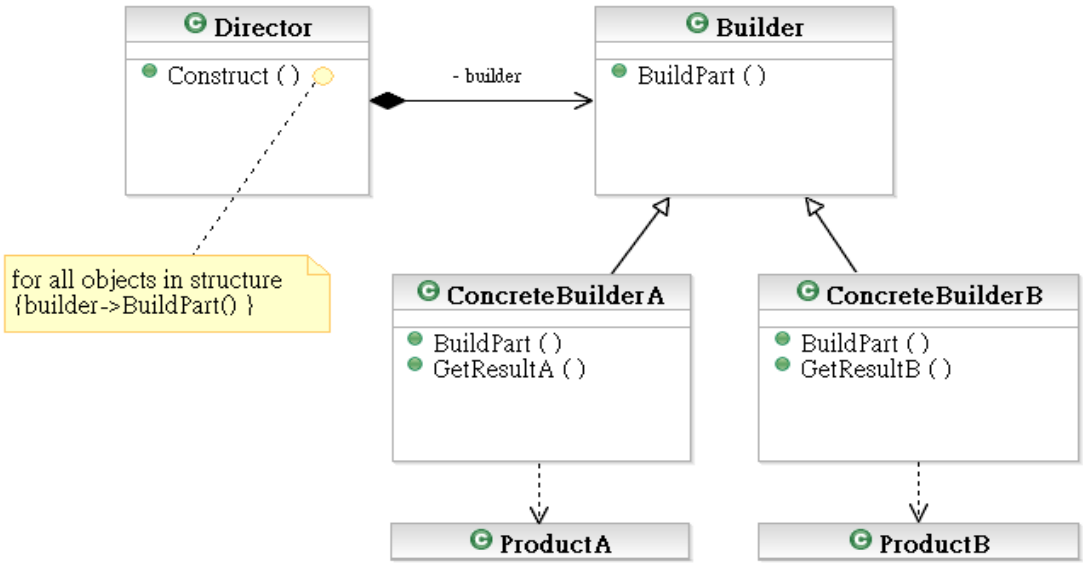
   (a) The input is an array A of size n. The output d is 1 if all elements of A are distinct; otherwise, the output d is 0.

   (b) The inputs are two sorted arrays A and B of sizes m and n respectively. Assuming that A and B have common elements, the output x is the smallest common element of A and B.

4. Prove the (partial) correctness of the following program. (20 %)

$\{(x = n) \wedge (n \geq 0)\}$
S1: $y := 0$;
S2: **while** $x > 0$ **do**
    S3: $y := y + (2x - 1)$;
    S4: $x := x - 1$
od
$\{y = n^2\}$

5. Describe as complete as possible in words what the following UML diagram is specifying: (10 %)



6. Suppose you are designing a programmable roving robot. The robot has the following classes for its sensor and controller:

    - `GPS`: a Global Positioning System class which supports the method `locate (double & longitude, double & latitude)`.
    - `Stepper`: a robot mobility unit which supports the method `move (double & dx, double & dy)`.
    - `Arm`: a robotic arm which supports the methods `retrieve (void)` and `deliver (void)`.

    Suppose the class `Robot` has the following definition:

```
class Robot {
  public:
    GPS gps;
    Stepper stepper;
    Arm arm;
    ...
};
```

Your goal is to define its programmable interface so that the robot can follow instructions programmed by users. Answer the following questions by writing pseudo C++ or Java code.

(a) Please use the Command pattern to define the class hierarchy and all classes that support the following instructions: (5 %)

- `GOTO X, Y`: goes to the absolute location (`X`, `Y`).
- `GET`: extends its arm to retrieve item(s).
- `PUT`: extends its arm to deliver item(s).

(b) Suppose we would like the robot to accept a sequence of instructions. How would you modify your design? (5 %)

(c) After programming the robot a while, we realize macro instructions such as `GET X, Y` and `PUT X, Y` are very useful. How would you add them in your design? (5 %)

(d) Consider the following sequence of instructions: `GET 0.0, 0.0`, `PUT 1.0, 1.0`, `GET 0.0, 0.0`, `PUT 2.0, 2.0`. Clearly, it is more efficient if the robot retrieves all items at once. That is, it executes the following instructions instead: `GET 0.0, 0.0`, `GET 0.0, 0.0`, `PUT 1.0, 1.0`, `PUT 2.0, 2.0`. Which design pattern would you use to make it smarter? Please modify your design properly. (5 %)

7. A simplistic car consists of an engine, a steering wheel, a brake, and wheels. Suppose you are designing a car controller which can be used in Ford Anglia of years 1962 and 1963.

(a) Among Abstract Factory, Builder, Factory Method, and Prototype patterns, which one would you choose in your design? Please explain your design decision in English or Chinese. (5 %)

(b) Please realize your design choice in pseudo C++ or Java code. (5 %)

(c) Since there can be at most four wheels in a car, please modify your code to ensure others would not create more wheels unwillingly. (5 %)

Appendix

- System LK: Axioms for Equality

  Let $t, s_1, \cdots, s_n, t_1, \cdots, t_n$ be arbitrary terms.

  $$\frac{}{\vdash t = t}$$

  For every n-ary function f,

  $$\frac{}{s_1 = t_1, \cdots, s_n = t_n \vdash f(s_1, \cdots, s_n) = f(t_1, \cdots, t_n)}$$

  For every n-ary predicate P (including $=$),

  $$\frac{}{s_1 = t_1, \cdots, s_n = t_n, P(s_1, \cdots, s_n) \vdash P(t_1, \cdots, t_n)}$$

  Note: The $=$ sign is part of the object language, not a meta symbol.