

『プログラム言語論』 演習課題

亀山

以下の課題について、pdf ファイルでレポートを作成し、manaba 経由で提出せよ。締め切りは、次回の授業の前日とする。質問は TA または、plm-report@logic.cs.tsukuba.ac.jp へにすること。

課題 1.(継承とサブタイピング) Java のプログラムが、`~kam/java/`に置いてある。Point.java, ColoredPoint.java, Test1.java, Test2.java というファイルは、一連のプログラムである。これを実行して、以下の質問に答えなさい。

Point クラスと ColoredPoint クラスは親子であり、Point が親、ColoredPoint が子である。Point クラスのメソッドを ColoredPoint クラスは継承 (inherit) するが、メソッド toString は別の定義に書き換えている。(いわゆる “override” である。)

Test1.java は、いろいろな test (test i, ii, ...) を含んでいる。(test i 以外はコメントになっているのでコメントにする記号をはずしてコンパイル、実行することにより調べなさい。ただし、いくつかのテストはコンパイルエラーになるので注意せよ。また、実行時エラーになるものもある。一方、コンパイル時の warning はここでは無視してよい。(ここでは、warning はコンパイルエラーとは見なさない。))

Test1.java のテストを参考に、以下の質問に答えなさい。

- (a) 親クラスのオブジェクトを、子クラスの変数に代入できるか？また、代入したあと、メソッド toString として、親の定義が使われるか、子の定義が使われるか。
- (b) 前問において、親と子を逆にするとどうなるか？
- (c) 親クラスのオブジェクトを、子クラスに cast (型変換) できるか？また、cast したあと、メソッド toString として、親の定義が使われるか、子の定義が使われるか。
- (d) 前問において、親と子を逆にするとどうなるか？
- (e) ColoredPoint の子クラス (Point クラスの孫クラス) として ColoredCircle を自分で定義して、上記 4 問を、Point と ColoredCircle の間で試しなさい。
- (f) (発展課題) 「祖父と孫」、「兄弟」などの間でどうなるか、などのパターンについても、自分で適当なクラスを定義した上で調べなさい。たとえば、兄弟とは、Point クラスの子クラス同士 (子クラス同士は特に関係がない) のことである。

課題 2 (overload と override) Test2.java では、foo というメソッドが 3 回定義されている。これらは引数の型あるいは引数の個数が異なっている。(いわゆる “overload” である。)

Test2.java のテストを実行してみて、foo の呼び出しが、それぞれどの定義を呼びだしているか答えなさい。また、この例を参考に、Java が、overload においてメソッドをどう決定するか、示しなさい。

なお、Test2.java は overload とサブタイピングが絡みあう、という、非常にややこしい例題も含んでいるので注意すること。

課題 3.(発展課題) (以下は、発展課題である。問題自体は全員が読んで理解してほしいが、解答は、余力がある人のみでよい。)

Test3a.java と Test3b.java はそれぞれ単独のテストプログラムである。これを実行して以下の間に答えなさい。(ただし、Test3a.java をコンパイルすると警告がでる。ここでは警告を気にしないものとする。)

- Java のリストは、要素の列である。ただし、まったく異なるクラスのオブジェクトを要素として持つのではなく、要素全体が同じクラスのオブジェクトか、そのサブクラスのオブジェクトであるべきである。たとえば、[1,2,3] はよいが [1,2,"abc"] は、(本来の意味からすると) リストと見なしたくない。Java が、そのような (おかしな) リストを許すかどうか、Test3a.java を参考にしして答えなさい。

- Test3a.java のようなリストの使い方をしていて、困った問題が生じる。それは、「整数のリスト」も「文字列のリスト」も区別なくすべて「(あらゆるクラスのオブジェクトを要素としてもつ) リスト」というクラスになってしまうため、型の整合性の検査が無意味になってしまうことである。テストプログラムでは Generics という機能を使って Integer リストや Point リストを実現して、この問題を回避している。テストプログラムを改造して、「Integer リストに、文字列を要素として追加」してみて、これが可能かどうか、調べなさい。
- (発展課題) テストプログラムの後ろの方の例題 (moveList の呼び出し等) は、1つのリストに、親子関係にあるオブジェクトを格納したときに、何らかのメソッドを呼び出すと、実際に呼びだされるのはどのメソッドか (親なのか子なのか、それはどうやって決まるか) をテストしている。これについて調べなさい。