COMPUTER SCIENCE TRIPOS Part IB, Part II 50% – 2022 – Paper 7

1 Concepts in Programming Languages (am21)

This question focuses on concepts – exact syntactic encoding is unimportant.

(*a*) Explain two possible implementation behaviours of the following program

```
let test(n:int) = (let f = (λx:int.n+x); n++; return f(10))
```

Now, give a corresponding Java-like program with `test` being a method containing a lambda and explain how Java resolves this situation. [4 marks]

(*b*) Explain two possible implementation behaviours of the following program

```
int n = 0; int g() = n; int f(int n) = g(); print(f(1));
```

briefly indicating how `g` would be implemented in each case. [4 marks]

(*c*) ML-family languages generally restrict polymorphic exceptions. Consider:

```
exception E of ('a->'a);;
try  raise (E(fun x->x))  with  E(f) -> (f 1, f true);;
```

Giving reasons: (*i*) would this code type-check in an ML-family language? (*ii*) would it execute successfully? Give a modified version of the code *with similar structure* (e.g. retaining two separate function applications to `1` and `true`) but which both type-checks and successfully evaluates to `(1,true)`. [5 marks]

(*d*) A blog proclaims "prototypes and virtual method tables are alternative implementations of inheritance". Clarify what was intended. [3 marks]

(*e*) Explain, with reasons, how much a Java compiler can optimise `CreateVec`, given that its definition and its calls may appear in separate Java source files

```
class IPair { final int x,y; IPair(int X,int Y) { x=X;y=Y; }
                /* other methods */    };
static IPair[] CreateVec(int n) { var V = new IPair[n];
  for (int i=0; i<n; i++) V[i] = new IPair(0,0);
  return V; }
```

[4 marks]