Compiler Construction
Supervision 3

Supervisor: Joe Isaacs (josi2).

All work should be submitted in PDF form 36 hours before the supervision to the email `josi2@cam.ac.uk`. If you have any questions on the course please include these at the top of the supervision work and we can talk about them in the supervision.

1. What is the job of a linker?
2. How could we OO classes with single inheritance. Describe the memory layout and how we could implement virtual methods.

   Use these classes as an example

   ```
   class P { int a, int b; int f() {...} }
   class Q extends P { int c; int f() {...}; int g() {...} }
   ```

3. `http://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2012p3q5.pdf`
4. `http://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2011p3q4.pdf` (c,d)
5. Give three options for handling memory for each given list advantages and disadvantages.
6. Given that we decide to use trace GC, what two methods could be used after tracing to free memory list advantages and disadvantages.
7. Practical exercise: `https://www.cl.cam.ac.uk/teaching/1920/CompConstr/practical.txt`

   (a) *The current compiler translates let-bindings as*
   let x = e1 in e2 end − − − > (fun x − > e2 end)e1
   This was nice in that we only had one form of abstraction to worry about. But it is not very efficient for most code. Imagine a function defined as
   fun f(x) = let x1 = e1 in let x2 = e2 in ... let xk = ek in e end ... end
   Work through how many closures get created an how things are computed at runtime. Arg! What to do? Hint : put the values of these local variables”in the stack frame of f! If the arg x to f is belowthe control entries (return address, pointer to f, saved frame pointer), then push the values of the x_i äbovethe control entries. You can now find their values by an index off of the current frame pointer. Now compare the runtime of this implementation with the closure-based approach ...

   You need not compare runtime.