CS558 Programming Languages – Fall 2023 – Study Questions Lecture 3b

These questions are intended for self-study, to help review and deepen your understanding of the lecture. Sample answers are available. There is nothing to hand in.

1. Storage *fragmentation* is a problem that occurs when there are enough total bytes of free memory to accommodate an allocation request, but they are not *contiguous*—so the allocation request fails. Explain (perhaps with examples) why fragmentation can be a problem in the heap if we use explicit deallocation operations and no garbage collection, but never in the stack.

2. The C# language supports both boxed and unboxed records: variables of `class` types ("reference" types in C# terminology) are boxed, but variables of `struct` types ("value" types in C# terminology) are unboxed. In other respects, `class`es and `struct`s are very similar: they have constructor methods, are created using the `new` operator, etc.

Based on this information, what is the output of the following C# program?

```
using System;

public class CL {
  public int A;
  public int B;

  public CL(int a, int b) { A = a; B = b; }
}

public struct ST {
  public CL C;
  public int D;

  public ST(CL c,int d) { C = c; D = d; }
}

public class Program {
  public static void Main() {
    CL cl = new CL(1,2);
    ST st1 = new ST(cl,3);
    ST st2 = st1;
    st2.D = 33;
    cl.A = 13;
    Console.WriteLine("st1.C.A = " + st1.C.A + " st1.C.B = " + st1.C.B + " st1.D = " + st1.D);
    Console.WriteLine("st2.C.A = " + st2.C.A + " st2.C.B = " + st2.C.B + " st2.D = " + st2.D);
 }
}
```

3. Using the information on slide 22, give a code example in Scala illustrating that structural equality does not in general imply reference equality.

4. (a) Show how to use pairs, as described on slide 23, to represent records with four integer fields. Using your representation, use infix dot notation to write the pair-based structure corresponding to the record containing (1,2,3,4), and draw the corresponding tree.

(b) Using your 4-tuple encoding from part (a) and the list encoding on slide 24, construct the three-element

list whose first element is (1,2), second element is 3, and third element is (4,5,6,7). Write this in infix dot notation and draw the corresponding tree.