TD12 : Compiling OOLanguages - Tuesday, December 14

Exercise 1 Consider the following Java program :

```
class A {
    int a = 0;
    int x() { return a; }
};

class B extends A {
    int b = 2;
    int e = 4;
    int y() { a = a - b + e; return a; }
};

class C extends A {
    int c = 3;
    int e = 5;
    int x() { return a + e; }
    int z() { e = a * c; return a; }
};

class Test {
     public static void main (String[] args) {
         A p = new B();
         B r = new B();
         C s = new C();
         A t = new A();
         p.x(); r.x(); s.x(); t.x();
     }
};
```

Answer the following questions :

Question 1 The program instantiates three classes of objects (A, B, and C). Show class hierarchy and the object layout.

Question 2 Can the `e` fields in classes `B` and `C` be placed at different offsets ?

Question 3 How does the compiler deal with polymorphism in this case ?

Question 4 How does the compiler deal with virtual methods ? How many different methods (i.e., assembly-level procedures) will be generated by the compiler for the above program ? What are the names of these methods ? Give their names in the "assemblerized" form classname methodname.

Question 5 Java permits a program to explicitly cast an object into another one. But there is a complication : the Java language requires a `ClassCastException` to be thrown, when the cast is not possible. What can the compiler do to allow for this possibility ?

Question 6 Draw the memory content at the end of main. Show the pointer links between the pointer variables (p, r, s, t), objects, dispatch tables, and procedures. Note : Your picture should have four kinds of

nodes (pointer variables, objects, dispatch tables, and procedures) and one kind of edge (denoting points-to relationship). The content of each pointer in the picture should be depicted as an edge to the target of the pointer.

Question 7 Give the sequence of assembler instructions that implements the dynamic dispatch call t.x(). Assume that the value of the variable t is stored in register R0. Assume that the VMTs and the CIRs are laid out as you defined above. Comment your code fragment : what does each offset mean ? what does each register contain ?

Exercise 2 Data Locality

Consider the code given below for computing a 2D Jacobi stencil :

```
for(int t=0; t<=T; t++)
    for (int i=1; i<=N-1; i++)
        for (int j=1; j<=N-1; j++)
            a[t][i][j] = 1/5*(a[t-1][i-1][j] +
                              a[t-1][i][j]   +
                              a[t-1][i][j+1] +
                              a[t-1][i+1][j] +
                              a[t-1][i][j-1]);
```

Question 8 Convert the above stencil code so to improve data locality using the method seen in course