1. We are given the following program in a new imperative programming language Tau. Its statements have the usual meanings.

```plaintext
main() {
    int x = 3;
    int y = 4;
    int z = 5;
    int w=10;

    f(){
        int z = 12;
        g(){
            int y=6;
            int z = 20;
            z=z+3;
            x++;  
            call h();
            print "g:\", z+x+y;
        }

        h(){
            x=x+2;
            y=y+4;
            z = z – w;
            print “h:\”, z+x+y  /******/
        }

        call g();
    }

call f();
}
```

a) What is the output of the program, if Tau is statically scoped? (8 pts)

b) What is the output of the program, if Tau is dynamically scoped? (8 pts)
c) Assume Tau is statically scoped, and static links are used to maintain scope information. Show the contents of the system stack at the point marked by /******/. Make sure you show all relevant pointers, the environment pointer EP, the pointer top, as well as the local variables, parameters, and other data stored in the activation records. (14 pts)

d) Assume that static scoping is used, with static links as the implementation of static scoping. The local offset of the variable "w" in function "main" is 20. How is the variable "w" represented in the function "h"? (hint: chain_offset, local_offset pair) (5 pts)
2) Assume that Tau is statically scoped, and we have the following Tau program.

```tau
main() {
    int x = 5;
    int y = 6;
    int z = 7;
    int w = 8;

    f(z) {
        y = y + x;
        w = w + 10;
        call g(y);
        print "f: ", x + y
    }

    g(int z) {
        z = z + w;
        print "g: ", z;
    }

    call f(w);
    print "main:", w
}
```

(note: "print" displays its parameter, and then a new line)

What is the output of the program if Tau uses the

a) By-value parameter passing mechanism? (9 pts)

b) By-reference parameter passing mechanism? (9 pts)

c) By-value-result parameter passing mechanism? (9 pts)
3) Assume we have the following class definitions in the object-oriented programming language T++. Assume all method calls are bound dynamically in T++.

```plaintext
class P {
    method m(){ ...} // address 100
    method n() {.....} // address 200
}

class C subclass_of P {
    method k() {....} // address 700
    method n(){.....} // address 300
    method s(){....} // address 400
}

class D subclass of C {
    method n(){……} // address 500
    method q(){…….} // address 800
}
```

Show the virtual method table for the class P. (5 pts)

Show the virtual method table for class D. (5 pts)
4) Given the following Haskell program,

\[
\begin{align*}
myst\ m\ [1] &= m \\
myst\ m\ [0] &= 0 \\
myst\ m\ (0:r) &= \myst\ (2*m)\ r \\
myst\ m\ (1:r) &= m + (\myst\ (2*m)\ r)
\end{align*}
\]

what is the value of the expression \( \myst\ 2\ [1,0,1,1] \) ? (5 pts)

______________________________

5) Given the following Haskell program,

\[
mystery\ x\ y = [z*2| z<- [x..y]]
\]

what is the value of the expression \( \mystery\ 2\ 5 \) ? (5 pts)

______________________________
6) True/False. Grading: 2 points for a correct answer, -1 point for an incorrect answer, 0 point for no answer.

   a. Constructors are called automatically just after an object is created ______________

   b. C is a pure object-oriented language. ______________

   c. An abstract class can be instantiated ______________

   d. Two primary features of ADTs are the packaging of data with their associated operations and inheritance ______________

   e. Instance methods have an extra “this” parameter ______________

   f. Multiple inheritance allows a new class to inherit from two or more classes ______________

   g. The encapsulation construct of ADA is called a package. ______________

   h. A coroutine is a special subprogram with multiple entries. ______________

   i. In a loop guarded command, if more than one condition is true, this results in an error situation. ______________

****************** Good luck!!!! ******************