Lecture 17, 60–100

ANSWERS TO

CLASS TEST#2 1996

OPERATIONS ON DATA

(1) What is the type of the program `g`

```
g x y z = x ++ y, if y!0 = 3
g x y z = z     , otherwise
```

(2) What is the closure of the set **{3, 6}** under the operation of division (/) ?

(3) Let `r = {(aaa,5),(bbb,4),(ccc,3)}`
`s = {(x,4),(y,6),(z,1),(y,3)}`

What is the result of:

```
project_first_of_3
 (join_second_of_2_with_second_of_2
        r (select_first_of_2 'y' s))
```

RECURSION

(4) What is wrong with this recursive definition:

```
f x y = f (x - y)    y  , if x < y
f x y = f   x     (y - x), if x >= y
```

(5) Let

```
h x []     =[]
h x (y:ys)=(x:(h x ys))++(x:(h x ys))
```

What is the value of `h 6 [1,2]`

(6) Use **RECURSION** to define a program `p` which takes a list of lists and a number **n** as input and which returns the number of lists in the input whose length is greater than **n** eg

```
p [[12,3,4],[3,4],
   [6,7,8,9],[3,4,12],[1]] 2 => 3
```

(7) Let the grammar `G` be defined as follows:

```
terminals = {0, 1, *},
non-terminals = {code},
distinguished non-terminal = code
   code ::= 0  | *
          | 0 code code | 1 code
```

Show how the expression `010*` can be derived

(8) Let the grammar `G` be defined as follows:

```
terminals = {0, 1, *},
non-terminals = {code},
distinguished non-terminal = code
   code ::= 0  | *
            | 0 code code | 1 code
```

Give a parse tree for the expression `0100`

(9) Let the grammar `G` be defined as follows:

```
terminals = {0, 1, *},
non-terminals = {code},
distinguished non-terminal = code
   code ::= 0  | *
            | 0 code code | 1 code
```

Show that the grammar G is ambiguous

SEMANTICS OF LANGUAGE

(10) Write down an attribute grammar for

```
L ={[3,2,5,6,4] => 4
    [8,6,4]     => 6,
    [10,4,2,4]  => 5,etc.
```