Please Remember:

- The course project is worth 25% of the grade.
- We recommend that you work in groups of up to two people.
- Project presentations are due.

Project Suggestions

1 Grammars

Given a grammar lie the following grammar

```
data Exp = IntConst Int
         | BoolConst Bool
         | Var Ident
         | App Exp Exp
         | Func Ident Exp
         | Cond Exp Exp Exp
         | Prim PrimId [Exp]
         | Let [(Ident, Exp)] Exp
         deriving (Eq,Text)
data Ident = Varid Int deriving (Eq,Text)
data PrimId = Add | Sub | Mul | Div | Eq | Less | Greater
              deriving (Eq,Text)
```

Write a program to compute the Normal Form of an expression in the presence of free variables.

2 The “Nine 9s” Problem

Write a program to determine the smallest positive integer that cannot be expressed as an expression composed of maximum of nine 9s and a small number of the operators +, -, *, /.

Hints:

Prefix	Data
7.14.*.*	A
7.14.7.3	B
10.18.200.*	C
10.18.200.5	D
5.*.*.*	E
*	F

Table 1: Example Table.

- The answer isn't zero. You can express zero like this: (9 - 9) * (9 + 9 + 9 + 9 + 9 + 9 + 9) Also, zero isn't a positive integer.
- The answer isn't one. You can express one like this: 9 - (9 * 9 - 9)/9 + 9 - 9 + 9 - 9
- It's not a trick question.
- Be sure to handle parentheses correctly.

Notes:

- You cannot exponentiate.
- You cannot concatenate (for example, put two 9s together to make 99).
- The - operator can be used in either its binary or unary form.
- Assume base 10.

3 The IP Lookup Problem

We suggest that you implement fast IP lookup in BlueSpec. The data structure for with this problem is an IP lookup table, which contains IP prefixes and associated data. The problem, hence, is to return the data associated with the *longest prefix match* ("LPM") for a given IP address (Tables 1 and 2).

Table representation issues:

- LPM is used for CIDR (Classless Inter-Domain Routing)
- Number of memory accesses for an LPM?
  - Too many → difficult to do LPMs at line rate

IP Address	Result
7.13.7.3	F
10.7.12.15	F
10.18.201.5	F
7.14.7.2	?
5.13.7.2	?
8.0.0.0	?
10.18.200.7	?

Table 2: Example Lookups.

IP Address	Result	M Ref
7.13.7.3	F	2
10.18.201.5	F	3
7.14.7.2	A	?
5.13.7.2	E	?
10.18.200.7	C	?

Table 3: Memory references when traversing the tree.

- Table size?
  - Too big → bigger SRAM → more latency, cost, power
- Control-plane issues
  - incremental table update
  - size, speed of table maintenance software

Figure 1 shows an example of a sparse tree representation for a lookup table given in Table 1. Table 3 depicts the number of memory references when traversing this sparse tree.

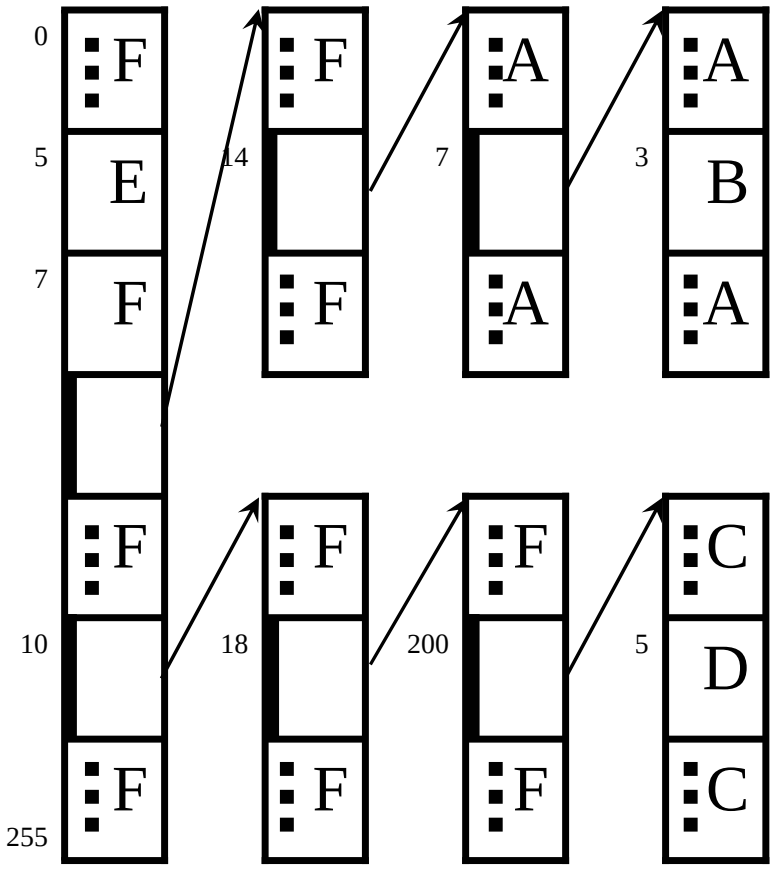

Figure 1: Sparse tree representation.