1. Current focus factor = 32/(3\*15) = 0.7111

0.7111 \* 4 \* 15 = 42.666  
0.7111 \* 1 \* 12 = 8.5332

Estimated velocity will be about 51.199.

1. You could estimate a new team’s focus factor by first figuring out how many days each individual person is going to be working within your sprints and find their individual focus. Someone working a 20 day sprint with 5 days off has an individual focus factor of 75%. Once you’ve computed all the individual focus factors you can then take the average of it to get your estimated focus factor for the new team.
2. You could have boxes of varying sizes and a number of apples or some such item. For each story you could place an amount of fruit into the box to represent the amount of point values it has. It’s not necessarily an elegant solution or easy to carry out but it could be a valuable way to represent the thought needed going into the task at hand. I don’t believe if would be much better than poker unless the team was far more visually oriented overall and the physical size of the boxes and fruit in them gave some sense to the tasks.
3. See diagram
4. package bianary.tree;

/\*\*

\*

\* @author Zac

\*/

public class BianaryTree {

node root;

BianaryTree(int input){

root = new node(input);

}

BianaryTree(){

root = null;

}

public static void main(String[] args) {

BianaryTree tree1 = new BianaryTree();

tree1.root = new node('1'); //creates the root

tree1.root.left = new node('2'); // creates left chile which makes 1 the parent

tree1.root.right = new node('3'); //creates the right child of 1

}

}

class node{

int key;

node left;

node right;

public node(int input){

key = input;

left = right = null;

}

}

f) see diagram

g)

public class Node {

private String name;

private Node next;

private Node prev;

public Node( String n, Node nex, Node p) // node class constructor

{

name = n;

next = nex;

prev = p;

}

public Node getNext() // returns next

{

return next;

}

public Node getPrev() // returns prev

{

return prev;

}

public String getName() // returns name

{

return name;

}

public class List {

Node next = new Node();

public void createList() // makes list

{

String name;

int SocialNum;

int Salary;

}

public static void main(String[] args) //driver

{

Linkedlists list = new Linkedlists();

list.createList();

}

}