Name:Hanliang Jiang
NetID:hj33

Problem 1:

1. Customers of AWS Lambda only needs to submit code and let Lambda automatically manage resource, while AWS EC2 requires customers to manually configure the instances on the virtual machine provided by AWS EC2. In general, EC2 requires more management and is more complex.
2. Regarding Price: AWS Lambda charges customers base on actual occupation of resource, while AWS Spot let customers bid for their occupied instances.
3. Azure Functions and Google Cloud Functions
4. I would choose AWS EC2 when I have a complicated task because AWS EC2 can be more flexible when defining underlying infrastructure; I would choose AWS Lambda when I try to focus on events and applications because AWS Lambda provides convenient automatic resource allocation.

Name:Hanliang Jiang
NetID:hj33

Problem 2:

1. Amazon EC2 P3; Azure DC1s v3/DC1ds v3; Google NVIDIA A100 GPUs
2. Amazon EC2 G4ad/G4dn; AzureA1 v2; Google NVIDIA P4 GPUs
3. TPUs have better performance on machine learning tasks, while GPUs are not as good at machine learning tasks but can also render graphics.

Name:Hanliang Jiang
NetID:hj33

Problem 3:

```
Map1(a,b):
Output(a,(OUT,b))
Output(b,(IN,a))

Reduce1(x,y):
Collect Sout = set of all (OUT,*) values from y
Collect Sin = set of all (IN,*) values from y
If |Sin|>=10M, Output((x,y),YES)
Else, Output((x,y),NO)

Map2((x,y),FLAG): identical

Reduce2((x,y),FLAG):
Collect Sout = set of all (OUT,*) values from y
Collect Sin = set of all (IN,*) values from y
If(|Sin|>=100M AND |Sout|<10 AND exists a value V from Sout whose FLAG==YES)
    Then output (x,_)
```

Name:Hanliang Jiang
NetID:hj33

Problem 5:

```
Map1(a,b):
Output(b,a);

Reduce1(x,V)
For each y in V
Lexicographic_sorted_pair((x,y),|V|)

Map2(a,b): identical
Reduce2(a,b):
If(|b|==1):Output nothing
If(|b|==2):Output (a,_)

Map3((a,b),_):
For each pair(a,b):
read (a, start_time_a, end_time_a)
read (b, start_time_b, end_time_b)
Output((a,b), (start_time_a, end_time_a, start_time_b, end_time_b))

Reduce3((a,b), (start_time_a, end_time_a, start_time_b, end_time_b)):
if (start_time_a - start_time_b) (end_time_a - end_time_b) > 0: Output(a,b)
else: Output nothing
```

Name:Hanliang Jiang
NetID:hj33

Problem 6:

```
Map(_, (V1, V2, V3)):
Output(V1, V2)
Output(V1, V3)
Output(V2, V3)
```

```
Reduce (a,b):
Collect V(i,j) = number of pair (i,j) in (a,b)
If(V(1,3)>V(3,1) AND V(1,2)>V(2,1))  Output(1,_)
Else If(V(2,3)>V(3,2) AND V(2,1)>V(1,2))  Output(2,_)
Else If(V(3,1)>V(1,3) AND V(3,2)>V(2,3))  Output(3,_)
Else Collect |b| Output (a,_) where a has biggest |b|
```

Name:Hanliang Jiang
NetID:hj33

Problem 7:
Let’s say for round I, the proportion of uninfected nodes is Pi, each node pull b nodes out of N, the probability that node remains uninfected is Pi^b, so the proportion of uninfected nodes will be $P_{i+1} = N * P_i * P_i/N = P_i^{b+1}$.After r rounds from beginning, $P_r = P_1^{(b+1)^r}$

The result of this equation is $r = log_{b+1} log \frac{P_r}{P_1}$.

Name:Hanliang Jiang
NetID:hj33

Problem 8:

Yes. (i)for an infected node, at the first stage, it takes O(logN) to disseminate inside targets; thereafter if it disseminate inside, one infected node disseminates b/(1-1/√ N) each round, and it takes O(logN) to disseminate all inside targets. If it disseminates outside, as there are √ N infected nodes, it takes O(1) to disseminate at least one node of its neighbour. In this pattern, all √N subnets can be infected O(logN).

(ii)Assume all nodes are infected, as each subnet Si chooses S(i+1), each router takes √N*(1/√N) = O(1) messages each round.

Name:Hanliang Jiang
NetID:hj33

Problem 9:
(i)advantage: can work on simultaneous multiple failures.
Disadvantage: if T(cleanup) = 0, neighbor would send a failed node after it timed out even though the node has failed. So no failed nodes can be detected.

(ii)advantage: constant time and load
disadvantage: removing suspicious feature cannot guarantee accuracy of detection. One working node with network problems can be marked as failed.

(iii)advantage: constant time and load
disadvantage: cannot guarantee completeness of detection, one failed node can take infinite time to be detected because of random picking