Eric Yang

304263623

CS 180

Dis 1B

Homework 1

1. a. For a favorable table, we find the leftmost column with odd number of 1’s and choose a row that has its most significant bit in that column. Changing all the 1’s to 0’s of that row for every column with odd number of 1’s will always make the table unfavorable.

For an unfavorable table, we know that each column has an even amount of 1’s. Thus if we draw matches from any one of the piles, we can either add a 1 or remove a 1 in one row only. Doing either action is guaranteed to cause at least one column to have an odd number of 1’s and result in a favorable table for the opponent.

If table is favorable

Choose row such that most significant bit is in same column as leftmost column with odd number of 1’s

For each 1 in row

If column has odd number of 1’s

Change 1 to 0

b. On every turn, pick matches carefully to ensure that there is an even number of 1’s in each column so that the opponent will always have an unfavorable table. And since we proved that for any unfavorable table, any move will make the table favorable for the opponent, we always get favorable tables and are guaranteed victory. This strategy only works assuming we don’t start with an unfavorable table and the opponent doesn’t know this strategy as well.

1. a. for each coefficient a

n = coefficient number

temp = 1

while n is not 0

temp = temp \*v

total = total + a \* temp

return total

this algorithm uses O(n^2) multiplications and additions

b.F()

total =

loop = n

while loop is not 0

total = total \* v

total = total + F()

return total

1. Recursive:

F(a,b)

If a < 10b

Use given algorithm

Return result of algorithm

else

Return F(a-b,b) +1

Iterative:

n = a

counter = 0

While n>=10b

n = n-b

counter = counter + 1

use given algorithm

return result of algorithm + counter

1. F(n)

Split players in half by low number and high number

F(n/2) lower half

F(n/2) higher half

Low number players will be matched with high number players on day n/2

Rotate the high number players for each following day until day n-1

Return table