University of Minnesota, School of Mathematics

Math 4707: Combinatorics, Spring 2018 Midterm 1

Brady Olson (edited by Darij Grinberg)

January 10, 2019

1 Exercise 1

1.1 Problem

Let $n \in \mathbb{N}$.

(a) Prove that the integer $\dbinom{2^n - 1}{b}$ is odd for each $b \in \{0, 1, \ldots, 2^n - 1\}$.

(b) Prove that the integer $\dbinom{2^n}{b}$ is even for each $b \in \{1, 2, \ldots, 2^n - 1\}$.

[Here, the set $\{0, 1, \ldots, 2^n - 1\}$ means the set of all integers k with $0 \leq k \leq 2^n - 1$, and the set $\{1, 2, \ldots, 2^n - 1\}$ means the set of all integers k with $1 \leq k \leq 2^n - 1$.]

1.2 Solution

Lemma 1.1. *The definition of binomial coefficient demonstrates that for any $k \in \mathbb{N}$*

$$\binom{0}{k} = [k = 0] .$$

This was shown briefly in class.

We now solve the actual exercise.

(a) *Proof.* Let $p(n)$ be the logical statement that **(a)** holds for some $n \in \mathbb{N}$. First, for $n = 0$ we have $b \in \{0, \ldots, 2^0 - 1\} = \{0\}$ so

$$\binom{2^n - 1}{b} = \binom{0}{0} \underset{Lemma\ 1.1}{=} 1,$$

which is clearly odd, so $p(0)$ is true.

For an inductive hypothesis, assume $p(m)$ holds for some $m \in \mathbb{N}$. Let's examine $p(m+1)$. We can express $2^{m+1} - 1$ as $2^m + (2^m - 1)$. Clearly $2^m - 1$ is in $\mathbb{N}$ since $2^m \in \mathbb{N}$ and $2^m \geq 1$. Fix $b \in \{0, 1, \ldots, 2^{m+1} - 1\}$.

If $b \leq 2^m - 1$, we are able to apply HW1 Exe4 (congruence 1) substituting $n := m,\ a := 2^m - 1,\ b := b$ since our variables satisfy the domain constraints (since $b \in \{0, 1, \ldots, 2^m - 1\}$), and obtain

$$\binom{2^{m+1} - 1}{b} = \binom{2^m + 2^m - 1}{b} \equiv \binom{2^m - 1}{b} \bmod 2.$$

If $b > 2^m - 1$, we are able to apply HW1 Exe4 (congruence 2) substituting $n := m,\ a := 2^m - 1,\ b := b - 2^m$ where our variables again satisfy the domain constraints (since $b > 2^m - 1$ yields $b - 2^m \geq 0$ and thus $b - 2^m \in \{0, 1, \ldots, 2^m - 1\}$), and obtain

$$\binom{2^{m+1} - 1}{b} = \binom{2^m + 2^m - 1}{2^m + b - 2^m} \equiv \binom{2^m - 1}{b} \bmod 2.$$

In either case, we obtain

$$\binom{2^{m+1} - 1}{b} \equiv \binom{2^m - 1}{b} \bmod 2.$$

From our inductive hypothesis, we know that $\binom{2^m - 1}{b}$ is odd, so $\binom{2^{m+1} - 1}{b}$ is also odd since they are congruent modulo 2. Thus, $p(m+1)$ holds given $p(m)$. Hence, $p(n)$ holds for all $n \in \mathbb{N}$ via the Principle of Mathematical Induction. □

(b) *Proof.* Both 0 and b belong to $\{0, 1, \ldots, 2^n - 1\}$. Hence, we can apply HW1 Exe4 (congruence 2) directly (substituting $n := n,\ a := 0,\ b := b$) to get

$$\binom{2^n}{b} \equiv \binom{0}{b} \bmod 2 \underset{Lemma\ 1.1}{=} 0 \bmod 2.$$

So $\binom{2^n}{b}$ is even. □

Remark 1.2. For **(b)**, note that the set $\{1, 2, \ldots, 2^n - 1\}$ is the empty set for $n = 0$. This means that **(b)** is vacuously true for $n = 0$, since no there are no b that it would make a statement about.

---

2 Exercise 4

2.1 Problem

(a) Given six integers $a_1, b_1, c_1, a_2, b_2, c_2$ satisfying $0 \leq a_1 \leq b_1 \leq c_1$ and $0 \leq a_2 \leq b_2 \leq c_2$. How many lattice paths from $(0, 0)$ to (c_1, c_2) pass through none of the points (a_1, a_2) nor (b_1, b_2)?

(b) Given six integers a, b, c, A, B, C satisfying $0 \leq a \leq b \leq c$ and $0 \leq A \leq B \leq C$. How many c-element subsets S of $[C]$ satisfy $|S \cap [A]| \neq a$ and $|S \cap [B]| \neq b$?

2.2 Solution

Lemma 2.1. *Let $m \in \mathbb{Z}$ and $n \in \mathbb{Z}$ be such that $m + n \geq 0$. The number of lattice paths from $(0,0)$ to (m,n) is $\binom{m+n}{n} = \binom{m+n}{m}$.*

Proof. In the case when $m \in \mathbb{N}$ and $n \in \mathbb{N}$, this was proven in class. Thus, it remains to consider the other case. So, we assume that one of m and n does not belong to $\mathbb{N}$. In other words, one of m and n is negative. Then, the other one is smaller than $m+n$. Hence, both $\binom{m+n}{n}$ and $\binom{m+n}{m}$ equal 0 (since $m + n \in \mathbb{N}$). It remains to check that the number of lattice paths from $(0,0)$ to (m,n) is 0 as well. But this is clear: Since one of m and n is negative, there are no lattice paths from $(0,0)$ to (m,n) (because both coordinates can only increase along a lattice path, and therefore we can never get to negative coordinates if we start at $(0,0)$). □

We now solve the exercise:

(a) Rather than just solving the exercise as it is stated, we generalize it a little bit: We replace the requirement "$0 \leq a_1 \leq b_1 \leq c_1$ and "$0 \leq a_2 \leq b_2 \leq c_2$" by the (weaker) requirement "$0 \leq a_1 + a_2 \leq b_1 + b_2 \leq c_1 + c_2$". This will come in handy when we later deduce part **(b)** from part **(a)**.

Proof. Define the following sets

$$U = \{\text{all lattice paths from } (0,0) \text{ to } (c_1, c_2)\},$$

$$P_a = \{\text{paths from } (0,0) \text{ to } (c_1, c_2) \text{ passing through } (a_1, a_2)\},$$
$$P_b = \{\text{paths from } (0,0) \text{ to } (c_1, c_2) \text{ passing through } (b_1, b_2)\};$$

thus,

$$U \setminus (P_a \cup P_b)$$
$$= \{\text{paths from } (0,0) \text{ to } (c_1, c_2) \text{ passing through neither } (a_1, a_2) \text{ nor } (b_1, b_2)\}.$$

But the Inclusion-Exclusion Principle shows that

$$|U \setminus (P_a \cup P_b)| = |U| - |P_a| - |P_b| + |P_a \cap P_b|.$$

So, we just need to count each of these sets using Lemma 2.1. If m, n, p, q are integers satisfying $0 \leq p + q \leq m + n$, then the number of paths from $(0,0)$ to (m,n) passing through (p,q) is $\binom{p+q}{p}\binom{(m+n)-(p+q)}{m-p}$ (since we are independently choosing paths from $(0,0)$ to (p,q) and from (p,q) to (m,n), but Lemma 2.1 shows that there are $\binom{p+q}{p}$ options for the former and $\binom{(m+n)-(p+q)}{m-p}$ options for the latter). This generalizes to a formula for the number of paths from $(0,0)$ to (m,n) passing through k specified points; it is expressed as a product of of $k+1$ binomial coefficients (indeed, any lattice path must traverse the k points in the order of increasing sum of coordinates[1], so it breaks into $k+1$ smaller paths with known endpoints). Hence,

$$|P_a| = \binom{a_1+a_2}{a_1}\binom{(c_1+c_2)-(a_1+a_2)}{c_1-a_1};$$
$$|P_b| = \binom{b_1+b_2}{b_1}\binom{(c_1+c_2)-(b_1+b_2)}{c_1-b_1};$$
$$|P_a \cap P_b| = \binom{a_1+a_2}{a_1}\binom{(b_1+b_2)-(a_1+a_2)}{b_1-a_1}\binom{(c_1+c_2)-(b_1+b_2)}{c_1-b_1}.$$

We can then count the desired lattice paths:

$$\begin{aligned}|U\setminus(P_a\cup P_b)| &= |U|-|P_a|-|P_b|+|P_a\cap P_b| \\ &= \binom{c_1+c_2}{c_1} - \binom{a_1+a_2}{a_1}\binom{(c_1+c_2)-(a_1+a_2)}{c_1-a_1} \\ &\quad - \binom{b_1+b_2}{b_1}\binom{(c_1+c_2)-(b_1+b_2)}{c_1-b_1} \\ &\quad + \binom{a_1+a_2}{a_1}\binom{(b_1+b_2)-(a_1+a_2)}{b_1-a_1}\binom{(c_1+c_2)-(b_1+b_2)}{c_1-b_1}.\end{aligned}$$

□

(b) *Proof.* Let $a_1 = a$, $b_1 = b$, and $c_1 = c$.
Let $a_2 = A-a$, $b_2 = B-b$, and $c_2 = C-c$.
Clearly, $0 \le a_1+a_2 \le b_1+b_2 \le c_1+c_2$.

Let P_c be the set $U\setminus(P_a\cup P_b)$ from **(a)**.

Let us construct a bijection between P_c and Q_c, the set of all c-element subsets S of $[C]$ satisfy $|S\cap[A]\,| \ne a$ and $|S\cap[B]\,| \ne b$.

Every path in P_c can be bijectively mapped to a subset S of $[C]$ with $|S| = c$. For a given path $P \in P_c$, it has $C = c_1+c_2$ steps, either north or east. Let us number these steps $\{1,2,\ldots,C\} = [C]$. Define a subset $S \subset [C]$ as the set of the indices of eastern steps (so a path from $(0,0)$ to $(1,1)$ consisting of an eastern step followed by a northern step would have $S = \{1\}$). This constructed set S is clearly a subset of $[C]$ containing $c = c_1$ elements. Since P does not pass through (a_1,a_2), P either takes more or less than a_1 eastern steps in its first $A = a_1+a_2$ steps. This implies that there are either more or less than $a = a_1$ elements in the first A elements of S, so $|S\cap[A]| \ne a$. Similarly, since P does not pass through (b_1,b_2), we have $|S\cap[B]| \ne b$. Thus, S is a valid member of Q_c.

The inverse of this function ($Q_c \to P_c$) follows naturally: For a given subset $S \in Q_c$, map each element $i \in S$ to an eastern step in a path P and each element $j \in [C]\setminus S$ to a northern step. A matching argument can be given to the correctness of this mapping. Moreover, this is a bijection since it is a two-sided inverse and each path P uniquely determines a subset S and vice versa.

Thus, there exists a bijection between P_c and Q_c, so that $|Q_c| = |P_c|$. The right hand side has been expressed in **(a)**. □

---

[1]If two of the k points have the same sum of coordinates, while being distinct, then the number is 0, because no lattice path can traverse them both.

3 Exercise 5

3.1 Problem

We say that a binary n-string $(a_1, a_2, \ldots, a_n)$ is *zig-zag* if it satisfies $a_1 \leq a_2 \geq a_3 \leq a_4 \geq \cdots$ (in other words, $a_i \leq a_{i+1}$ for every odd $i \in [n-1]$, and $a_i \geq a_{i+1}$ for every even $i \in [n-1]$).

Find a simple expression (no summation signs, only known functions and sequences) for the number of zig-zag binary n-strings for all $n \in \mathbb{N}$.

3.2 Solution

We shall use the ceiling function: For any integer x, we let $\lceil x \rceil$ denote the smallest integer that is $\geq x$.

First let us define the color of a binary string b_n as

$$color(b_n) = \begin{cases} \text{black} & \text{if } b_n\text{'s final element is } 0 \\ \text{red} & \text{otherwise} \end{cases}$$

(in particular, we count the binary 0-string () as red). Let $z_{n,black}$ and $z_{n,red}$ denote the number of black and red zig-zag binary n-strings, respectively. Let z_n be the number of all zig-zag binary n-strings. Then, $z_n = z_{n,black} + z_{n,red}$, since every zig-zag binary string must either be black or red.

If b is a binary n-string, and if $g \in \{0, 1\}$, then $b\#g$ will mean the binary $(n+1)$-string obtained by appending g to the end of b. (For example, $(0,1,1,0)\,\#1 = (0,1,1,0,1)$.)

Let us devise a constructive algorithm to recursively build zig-zag binary strings. Given a zig-zag binary n-string b_n, the following algorithm constructs all zig-zag binary $(n+1)$-strings that begin with b_n:

Algorithm 3.1 Generating zig-zag binary strings

```
1: procedure GenerateZigZag(b_n)
2:     if b_n is black then
3:         if n is even then return b_n#0
4:         else return b_n#0, b_n#1
5:     else    % b_n is red
6:         if n is even then return b_n#0, b_n#1
7:         else return b_n#1
```

This algorithm can be shown to be correct and that it generates *all* zig-zag binary $(n+1)$-strings. Simply, it generates all binary $(n+1)$-strings but prunes those that do not satisfy the zig-zag condition; lines 3 and 7 exclude strings which do not satisfy $a_i \leq a_{i+1}$ for every odd $i \in [n]$, and $a_i \geq a_{i+1}$ for every even $i \in [n]$.

Applying Algorithm 3.1 for low n gives familiar values for $z_{n,black}$, $z_{n,red}$, and z_n:

We now solve the exercise:

Proof. Define $p(n)$ as the logical statement, for $n \in \mathbb{N}$, that

$$z_{n,black} = f_{2\lceil \frac{n}{2} \rceil} \qquad \text{and} \qquad z_{n,red} = f_{2\lceil \frac{n+1}{2} \rceil - 1},$$

where f_i is the i^{th} Fibonacci number (having $f_0 = 0, f_1 = 1$). For the base case, $n = 0$, we have one red zig-zag binary string and no black zig-zag binary strings, so $p(0)$ holds since $z_{n,black} = f_{\lceil 2\frac{0}{2} \rceil} = f_0 = 0$ and $z_{n,red} = f_{2\lceil \frac{0+1}{2} \rceil - 1} = f_1 = 1$.

n	$z_{n,black}$	$z_{n,red}$	z_n
0	0	1	1
1	1	1	2
2	1	2	3
3	3	2	5
4	3	5	8
5	8	5	13
$\vdots$	$\vdots$	$\vdots$	$\vdots$

For the inductive hypothesis, assume $p(m)$ holds for some $m \in \mathbb{N}$. Then, let's examine $p(m+1)$; count the red and black zig-zag binary $(m+1)$-strings.

If m is even, for each black zig-zag binary m-string, GenerateZigZag will generate 1 black zig-zag binary $(m+1)$-string (line 3). Likewise, each red zig-zag binary m-string will generate 1 black zig-zag binary $(m+1)$-string and 1 red zig-zag binary $(m+1)$-string (line 6). So, there are $z_{m,black} + z_{m,red}$ black zig-zag binary $(m+1)$-strings and $z_{m,red}$ red zig-zag binary $(m+1)$-strings. This gives us the expressions

$$z_{m+1,black} = z_{m,black} + z_{m,red} \underset{IH}{=} f_{2\lceil \frac{m}{2} \rceil} + f_{2\lceil \frac{m+1}{2} \rceil - 1}$$

(the symbol "$\underset{IH}{=}$" means "equals, by the inductive hypothesis"). Since m is even, this evaluates to

$$z_{m+1,black} = f_m + f_{m+2-1} = f_{m+2} = f_{2\lceil \frac{m+1}{2} \rceil},$$

where the last equality is true since m is even. More simply, the number of red zig-zag binary $(m+1)$-strings is the same as the number of those with length m:

$$z_{m+1,red} = z_{m,red} = f_{2\lceil \frac{m+1}{2} \rceil - 1} = f_{2\lceil \frac{(m+1)+1}{2} \rceil - 1},$$

where again the last equality holds since m is even.

On the other hand, if m is odd, for each black zig-zag binary m-string, GenerateZigZag will generate 1 black zig-zag binary $(m+1)$-string and 1 red zig-zag binary $(m+1)$-string (line 4). Likewise, each red zig-zag binary m-string will generate 1 red zig-zag binary $(m+1)$-string (line 7). So, there are $z_{m,black} + z_{m,red}$ red zig-zag binary $(m+1)$-strings and $z_{m,black}$ black zig-zag binary $(m+1)$-strings. A very similar argument as with even m (where ceiling terms are affected differently since m is odd) yields the same results.

So, $p(m+1)$ holds given $p(m)$. Hence, by the Principle of Mathematical Induction, $p(n)$ holds for all $n \in \mathbb{N}$.

Now, recall that $z_n = z_{n,black} + z_{n,red}$. Since $p(n)$ holds, this simplifies to

$$z_n = f_{2\lceil \frac{n}{2} \rceil} + f_{2\lceil \frac{n+1}{2} \rceil - 1} = \begin{cases} f_n + f_{n+1} & \text{if } n \text{ is even} \\ f_{n+1} + f_n & \text{if } n \text{ is odd} \end{cases} = f_{n+2}.$$

So, the number of zig-zag binary n-strings is $z_n = f_{n+2}$, the $(n+2)^{th}$ Fibonacci number. □