NUMBER OF SELF-AVOIDING LATTICE PATHS

AIDAN TAN, BRYAN LEE, CALEB SEOW, JOVAN LOW

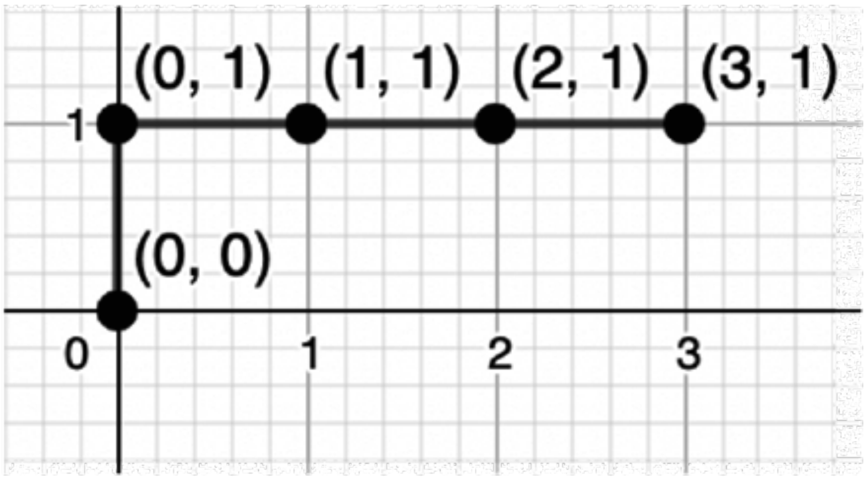
A. Introduction

i. Definitions

Self Avoiding Lattice Paths

In combinatorics, a self avoiding lattice path is a sequence of distinct points $p_0, p_1, p_2, \ldots$ in $\mathbb{Z}^d$, such that the difference between any 2 points gives a unique operation in some set S.

For instance, the set $\{(0,0),\ (0,1),\ (1,1),\ (2,1),\ (3,1)\}$ is a self avoiding lattice path from $(0,0)$ to $(3,1)$, with $S = \{(1,0),\ (0,1)\}$. The path it traces is shown below



Cartesian Products

The cartesian product of two sets, A and B, is denoted $A \times B$, and is defined as $A \times B = \{(a,b): a \in A,\ b \in B\}$

ii. Notation

We denote
$R_{n,m} = \{0, 1, 2, \ldots, n\} \times \{0, 1, 2, \ldots, m\}$

B. Case 1, North and East

Define $S = \{(1, 0),\ (0, 1)\}$.

The number for paths taken in an $R_{n,m}$ lattice is given by:

$$^{n+m}C_m = \ ^{n+m}C_n = \frac{(n+m)!}{n!\,m!}$$

In this section, we explore the base case for the counting of lattice paths, as well as build an understanding of the combinatorics formula to be applied in future sections.

As there are no diagonal lines, there must be n number of $(1, 0)$ steps and m number of $(0, 1)$ steps in an $R_{n,m}$ lattice. Thus, the total number of paths from $(0, 0)$ to (n, m) with steps in $S = \{(1, 0),\ (0, 1)\}$ is $n + m$.

What the formula does is it finds the number of ways n (or m) steps can be arranged in a total of $n + m$ steps.

We start by finding the number of permutations $n + m$ steps can be arranged, which is given by $^{n+m}P_{n+m} = (n + m)!$. By restricting the number of steps of (1,0) to n, we get $^{n+m}P_n = \frac{(n+m)!}{(n+m-n)!} = \frac{(n+m)!}{m!}$. Notice that this is different from when we restrict the number of steps of (0,1) to m, which gives us $^{n+m}P_m = \frac{(n+m)!}{(n+m-m)!} = \frac{(n+m)!}{n!}$.

Finally, as all n steps are identical (all of them being $(1, 0)$ in this case), we remove all repeated arrangements paths by dividing everything by n-permutations, or $n!$. This gives us $^{n+m}C_n = \frac{(n+m)!}{n!\,m!}$. Now, notice that by dividing the permutations of steps arranged to m by m-permutations, we also get $^{n+m}C_m = \frac{(n+m)!}{n!\,m!}$. Thus, by transitive property $^{n+m}C_n = \ ^{n+m}C_m = \frac{(n+m)!}{n!\,m!}$.

Hence, to find the total number of paths in a square lattice with operation set $S = \{(1, 0),\ (0, 1)\}$, we simply need to find the number of ways we can arrange n steps (or m steps), among the total number of $n + m$ steps. In other words, the formula is simply $^{n+m}C_n$.

Intuitively, this also explains why $^{n+m}C_n = \ ^{n+m}C_m$, as they are essentially finding the same things.

C. Case 2, North, East, and North-East

Define $S = \{(1,0), (0,1), (1,1)\}$.

We claim that the number of lattice paths from (0,0) to (n,m) with steps in S, and fall within $R_{n,m}$ is given by:

$$\sum_{i=0}^{\min(m,n)} \frac{(m+n-i)!}{i!\,(n-i)!\,(m-i)!}$$

Derivation:
Any lattice path of such form, can have any number of diagonal steps ranging from 0 to the smaller number between n and m. That is, letting i be the number of diagonal steps,
$0 \le i \le min(n,m)$

> Suppose n is the smaller of the two. If $i = n + 1$, then the final lattice point in the lattice path will at least have x-coordinate $n + 1$. This falls outside the boundary $R_{n,m}$. On the other hand, if $i \le n$, the path may fall in the boundary.

With i steps in $(1,1)$, the lattice path must take another $n - i$ steps in $(1,0)$, and $m - i$ steps in $(0,1)$, in order to arrive at (n,m). Notice this is because:

$$i \cdot (1,1) + (n-i) \cdot (1,0) + (m-i) \cdot (0,1) = (n,m).$$

Since these steps may occur in any order, a lattice path with our definition of S is simply any rearrangement of i $(1,1)$ steps, $n - i$ $(1,0)$ steps, and $m - i$ $(0,1)$ steps.

A rearrangement of those steps can be found as follows:

In a string of $i + (n-i) + (m-i) = m + n - i$ steps, choose i locations for $(1,1)$. In positions other than those containing$(1,1)$, choose $(n-i)$ locations for$(1,0)$. Finally, in the remaining positions, put $(0,1)$ steps.

The first "action" has ${}^{n+m-i}C_i$ number of ways to perform.
The second has ${}^{(n+m-i)-i}C_{n-i} = {}^{n+m-2i}C_{n-i}$ number of ways to perform.
The third has simply 1 way to perform.

Finally, multiplying them to find the total number of rearrangements, while also using the factorial definition of combinations gives us:

$$\binom{m+n-i}{i}\binom{m+n-2i}{n-i} = \binom{m+n-i}{m+n-2i}\binom{m+n-2i}{m-i} = \frac{(m+n-i)!}{(m+n-2i)!\,i!} \cdot \frac{(m+n-2i)!}{(n-i)!\,(m-i)!} = \frac{(m+n-i)!}{(n-i)!\,(m-i)!\,i!}$$

Summing over all possible numbers of i, we obtain:

$$\sum_{i=0}^{\min(m,n)} \frac{(m+n-i)!}{i!\,(n-i)!\,(m-i)!}$$

The result we wanted.

To check if this formula makes sense, notice that for any (m, n), a formula $P(n, m)$ giving the number of paths should satisfy the following recurrence relation:

$$P(n,m) = P(n-1,m) + P(n,m-1) + P(n-1,m-1)$$

Each path from $(n-1, m)$ contributes 1 path after adding a step of $(1, 0)$. This same logic applies for $(n, m-1)$ with $(0, 1)$ and $(n-1, m-1)$ with $(1, 1)$. Thus, the combined number of paths leading to these specific points must add up to the number of paths leading to (n, m).

This can be derived as (n, m) is connected to only these three other points in the lattice. Thus, any possible solution to reach (n, m) must have passed through one of these three points as the second last point passed before reaching to (n, m). Any specific solution in $P(n, m)$ must thus exist as a unique solution in one of the following: $P(n-1, m)$; $P(n, m-1)$; $P(n-1, m-1)$, with the respective motion of $(1, 0)$; $(0, 1)$; $(1, 1)$ added behind.

We thus utilise the recurrence relation to check our formula.

Without loss of generality, assume $n < m$ and that both are natural numbers. We wish to show:

$$\sum_{i=0}^{min(n,m)} \frac{(m+n-i)!}{i!\cdot(n-i)!\cdot(m-i)!} = \sum_{i=0}^{min(n-1,m)} \frac{(m+n-1-i)!}{i!\cdot(n-i)!\cdot(m-i)!} + \sum_{i=0}^{min(n,m-1)} \frac{(m-1+n-i)!}{i!\cdot(n-i)!\cdot(m-1-i)!} + \sum_{i=0}^{min(n-1,m-1)} \frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}$$

Since $n \le m-1$, and both are integers imply $n-1 < m$, $min(n,m) = min(n, m-1) = n$ and $min(n-1, m) = n-1$.

We examine the difference:

$$\sum_{i=0}^{n-1} \frac{(m+n-1-i)!}{i!\cdot(n-1-i)!\cdot(m-i)!} + \sum_{i=0}^{n} \frac{(m-1+n-i)!}{i!\cdot(n-i)!\cdot(m-1-i)!} + \sum_{i=0}^{n-1} \frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!} - \sum_{i=0}^{n} \frac{(m+n-i)!}{i!\cdot(n-i)!\cdot(m-i)!} = a$$

And by showing that this difference equals zero, we show the above equality (notice that each sum in the difference corresponds to a sum in the claim).

Taking out the last term of the second and last sum leaves:

$$\left(\sum_{i=0}^{n-1}\frac{(m+n-1-i)!}{i!\cdot(n-1-i)!\cdot(m-i)!}+\sum_{i=0}^{n-1}\frac{(m-1+n-i)!}{i!\cdot(n-i)!\cdot(m-1-i)!}+\sum_{i=0}^{n-1}\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}-\sum_{i=0}^{n-1}\frac{(m+n-i)!}{i!\cdot(n-i)!\cdot(m-i)!}\right)$$
$$+\frac{(m-1)!}{n!\cdot(m-n-1)!}-\frac{m!}{n!\cdot(m-n)!}=a$$

Which leads to:

$$\sum_{i=0}^{n-1}\left(\frac{(m+n-1-i)!}{i!\cdot(n-1-i)!\cdot(m-i)!}+\frac{(m-1+n-i)!}{i!\cdot(n-i)!\cdot(m-1-i)!}+\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}-\frac{(m+n-i)!}{i!\cdot(n-i)!\cdot(m-i)!}\right)$$
$$=a+\frac{m!}{n!\cdot(m-n)!}-\frac{(m-1)!}{n!\cdot(m-n-1)!}$$

Factoring out $\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}$ gives:

$$a+\frac{m!}{n!\cdot(m-n)!}-\frac{(m-1)!}{n!\cdot(m-n-1)!}=\sum_{i=0}^{n-1}\left(\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}\right)\left(\frac{m+n-1-i}{m-i}+\frac{m+n-1-i}{n-i}+1-\frac{(m+n-i)\cdot(m+n-i-1)}{(n-i)\cdot(m-i)}\right)$$
$$=\sum_{i=0}^{n-1}\left(1-\frac{i\cdot(m+n-i-1)}{(n-i)\cdot(m-i)}\right)\left(\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}\right)$$
$$=\sum_{i=0}^{n-1}\left(\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}-\left(\frac{i\cdot(m+n-i-1)}{(n-i)\cdot(m-i)}\right)\left(\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}\right)\right)$$

We now take out the 0th term of the sum, as this will lead to a telescoping sum (we wish to use the property $\frac{a!}{a}=(a-1)!,\ a\geq 1$):

$$a+\frac{m!}{n!\cdot(m-n)!}-\frac{(m-1)!}{n!\cdot(m-n-1)!}$$
$$=\sum_{i=1}^{n-1}\left(\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}-\frac{(m+n-i-1)!}{(i-1)!\cdot(m-i)!\cdot(n-i)!}\right)+\left(\frac{(m+n-2)!}{(n-1)!\cdot(m-1)!}-\frac{(0)\cdot(m+n-1)}{(n)\cdot(m)}\left(\frac{(m+n-2)!}{(n-1)!\cdot(m-1)!}\right)\right)$$
$$=\sum_{i=1}^{n-1}\left(\frac{(m+n-i-2)!}{i!\cdot(n-1-i)!\cdot(m-1-i)!}-\frac{(m+n-i-1)!}{(i-1)!\cdot(m-i)!\cdot(n-i)!}\right)+\left(\frac{(m+n-2)!}{(n-1)!\cdot(m-1)!}\right)$$
$$=\frac{(m+n-(n-1)-2)!}{(n-1)!\cdot(n-1-(n-1))!\cdot(m-1-(n-1))!}-\frac{(m+n-1-1)!}{(1-1)!\cdot(m-1)!\cdot(n-1)!}+\frac{(m+n-2)!}{(n-1)!\cdot(m-1)!}$$
$$=\frac{(m-1)!}{(n-1)!\cdot(m-n)!}$$
$$=\frac{(m-1)!}{(n-1)!\cdot(m-n)!}\cdot\frac{m-(m-n)}{n}$$
$$=\frac{m!}{n!\cdot(m-n)!}-\frac{(m-1)!}{n!\cdot(m-n-1)!}$$

Finally, cancelling off like terms on both sides, give us our wanted $a=0$, thus proving the identity. Notice that $n=m$ does not change the proof, as it only changes the upper index of the summations to $n-1$, which removes the term $\frac{(m-1)!}{n!\cdot(m-n-1)!}$, leaving the end equation as

$$a+\frac{m!}{n!\cdot(m-n)!}$$
$$=a+1$$

$$= \sum_{i=1}^{n-1} \left(\frac{(m+n-i-2)!}{i! \cdot (n-1-i)! \cdot (m-1-i)!} - \frac{(m+n-i-1)!}{(i-1)! \cdot (m-i)! \cdot (n-i)!} \right) + \left(\frac{(m+n-2)!}{(n-1)! \cdot (m-1)!} - \frac{(0) \cdot (m+n-1)}{(n) \cdot (m)} \left(\frac{(m+n-2)!}{(n-1)! \cdot (m-1)!} \right) \right)$$

$$= \sum_{i=1}^{n-1} \left(\frac{(m+n-i-2)!}{i! \cdot (n-1-i)! \cdot (m-1-i)!} - \frac{(m+n-i-1)!}{(i-1)! \cdot (m-i)! \cdot (n-i)!} \right) + \left(\frac{(m+n-2)!}{(n-1)! \cdot (m-1)!} \right)$$

$$= \sum_{i=1}^{n-2} \left(\frac{(m+n-i-2)!}{i! \cdot (n-1-i)! \cdot (m-1-i)!} - \frac{(m+n-i-1)!}{(i-1)! \cdot (m-i)! \cdot (n-i)!} \right) + \frac{(m+n-n+1-2)!}{(n-1)! \cdot (n-1-(n-1))! \cdot (m-1-(n-1))!}$$

$$- \frac{(m+n-(n-1)-1)!}{((n-1)-1)! \cdot (m-(n-1))! \cdot (n-(n-1))!} + \frac{(m+n-2)!}{(n-1)! \cdot (m-1)!}$$

$$= \sum_{i=1}^{n-2} \left(\frac{(m+n-i-2)!}{i! \cdot (n-1-i)! \cdot (m-1-i)!} - \frac{(m+n-i-1)!}{(i-1)! \cdot (m-i)! \cdot (n-i)!} \right) + \frac{(m-1)!}{(n-1)!} - \frac{m!}{(n-2)! \cdot (m-n+1))!} + \frac{(m+n-2)!}{(n-1)! \cdot (m-1)!}$$

$$= \sum_{i=1}^{n-2} \left(\frac{(m+n-i-2)!}{i! \cdot (n-1-i)! \cdot (m-1-i)!} - \frac{(m+n-i-1)!}{(i-1)! \cdot (m-i)! \cdot (n-i)!} \right) + 1 - \frac{m!}{(n-2)!} + \frac{(m+n-2)!}{(n-1)! \cdot (m-1)!}$$

$$= \frac{(m+n-2-(n-2))!}{(n-2)!\,(m-(n-2)-1)!} - \frac{(m+n-(1)-1)!}{(m-1)! \cdot (n-1)!} + 1 - \frac{m!}{(n-2)!} + \frac{(m+n-2)!}{(n-1)! \cdot (m-1)!}$$

$$= 1$$

Giving $a = 0$ as well.

D. Case 3, North, East, West, North-East, North-West

Define $S = \{(1,0), (0,1), (-1,0), (1,1), (-1,1)\}$.

We claim that the number of self-avoiding paths from $(0,0)$ to (n,m) with steps in S, and fall within $R_{n,m}$, is given by:

$$(3n+1)^m$$

Derivation:
For all self-avoiding paths from $(0,0)$ to (n,m) with the above definition, there must exist a point in the set of points defining the path that has y coordinate $(m-1)$.

> Suppose for instance this is false. Then there exists a path that skips from y coordinate $(m-z)$ to m for $z > 1$. That is for some points a_i, a_{i+1}, their difference has y coordinate z. Clearly this is not in our set of operations.

This shows that the number of paths from $(0,0)$ to $(n-m)$ has some relationship to the number of paths from the "row" below it. We will show that the relationship is

$$P(n,m) = (3n+1)\,P(n,m-1)$$

Where $P(n,m)$ is the number of paths from $(0,0)$ to (n,m) within $R_{n,m}$

(1) For any path from $(0,0)$ to $(i,m-1)$ within $R_{n,m}$, with $0 \le i \le n$, it may be extended to a path to (n,m) in either 2 or 3 ways.

This can be seen by the fact that for any point $(i,m-1)$ moving in either $(0,1)$, $(1,1)$ or $(-1,1)$ will reach the row with y coordinate m, at which there is only one distinct path from that point to(n,m).

However, at the borders, that is, at $(0,m-1)$ and $(n,m-1)$, moving in $(-1,1)$ and $(1,1)$ respectively will lead to the path falling outside $R_{n,m}$, therefore any path to either point may only be extended in 2 ways.

(2) The paths found above include all paths to $(n.m)$.

Suppose there exists a path not included in the above claim. Then, as established, it must come from some point $(i,m-1)$. From there, moving in $(0,1)$, $(1,1)$ or $(-1,1)$ will be included in claim (1), while moving in $(-1,0)$ or $(1,0)$ some number of times will only lead to a different point $(i+1,m-1)$ on the row with y coordinate $(m-1)$, where from there one must go up in any of the 3 directions to reach (n, m), and that specific path is included in the paths leading to (n, m) from $(i + a, m - 1)$, as we are considering all paths from $(0, 0)$ to $(i + a, m - 1)$ in $R_{m,n}$ in this case.

(3) The number of paths from $(0, 0)$ to 2 different points in the same row remain the same

Consider the number of paths from $(0, 0)$ to (i, m) and (p, m).

A path from $(0, 0)$ to (i, m) must have a first point on y coordinate m (k, m) in its sequence.

If $k = p$, then we may simply stop the path there to get a path to $(p. m)$. Note that we cannot move away and return back to this position as we would have to retrace our steps, going against the definition of "self-avoiding paths".
If $k < p$, then we may simply move $p - k$ times in $(1, 0)$ to reach (p, m).
If $k > p$, then we may simply move $k - p$ times in $(- 1, 0)$ to reach (p, m).

That is, any path from $(0, 0)$ to (i, m) is extendable to a path to (p, m), and vice versa (by exchanging p with i).

This shows claim (3), as the number of paths from $(0, 0)$ to (i, m) will be less than or equal to the number of paths from $(0, 0)$ to (p, m), and vice versa, leading to the conclusion that they are equal.

Finally,

$$\begin{aligned} P(n, m) &= \sum_{i=0}^{n} 3P(i, m - 1) - P(0, m - 1) - P(n, m - 1) && (1), (2) \\ &= 3\sum_{i=0}^{n} P(n, m - 1) - P(n, m - 1) - P(n, m - 1) && (3) \\ &= 3\,(n + 1)\,P(n, m - 1) - 2\,P(n, m - 1) \\ &= (3n + 1)\,P(n, m - 1) \end{aligned}$$

For m in the positive integers.

The above recurrence relation can be solved easily via induction. However, here we use generating functions. Notice that in the relation, n stays constant, although it can be chosen arbitrarily. Thus, let $a_m = ca_{m-1}$, where $a_m = P(n, m)$, $c = (3n + 1)$

Let $f(x) = \sum_{k=0}^{\infty} a_k x^k$

By recurrence,

$$cxf(x) = \sum_{k=0}^{\infty} a_{k+1}x^{k+1}$$
$$= \sum_{k=1}^{\infty} a_k x^k$$
$$= f(x) - a_0$$

Since $a_m = P(n, m)$, and the number of paths from $(0, 0)$ to $(n, 0)$ is simply 1 by our definition of the set of operations (the path of simply n number of $(1, 0)$ steps), we may set $a_0 = 1$, which gives:

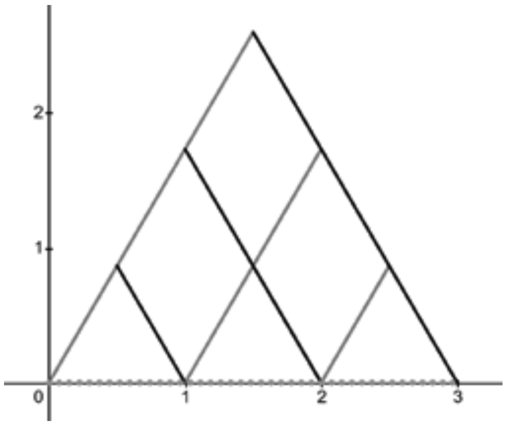
$$cxf(x) = f(x) - 1$$
$$(1 - cx)\, f(x) = 1$$
$$f(x) = \frac{1}{1-cx} = \sum_{k=0}^{\infty} c^k x^k$$

Where the last step involves a very common expansion for $|x| < 1$.
Matching coefficients of x, we obtain $a_k = c^k$, implying $a_m = (3n + 1)^m$, the result we were looking for.

One may also use a similar idea to show that the set $S = \{(1, 0), (0, 1), (-1, 0)\}$ has general number $(n + 1)^m$, which we will utilise later on, while the set $S = \{(1, 0), (0, 1), (-1, 0), (1, 1)\}$ has general number $(2n + 1)^m$.

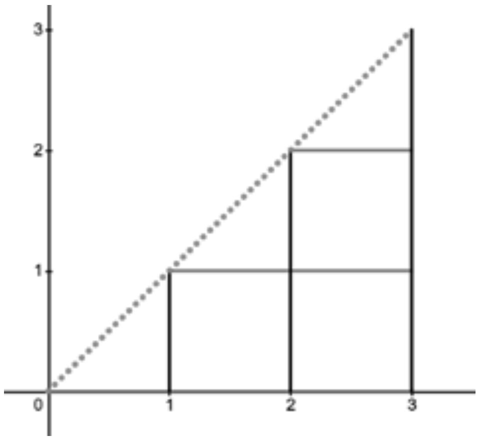
E. Case 4, Triangular Lattice, Up (Right), Down (Left)

In Case 4, we move away from the square grids that we have been using in Cases 1 to 3, to use a triangular grid as shown below, the aim being to find the number of paths from $(0, 0)$ to the right-most corner of the figure (in the example, $(3, 0)$):



That being said, despite the aesthetic of triangles, S would be defined as
$S = \{(cos\frac{\pi}{3}, sin\frac{\pi}{3}), (cos\frac{\pi}{3}, sin\frac{4\pi}{3})$.

In an attempt to make things clearer, we can transform the entire figure into a right angled triangle, with the right angle located at $(n, 0)$, with the endpoint at (n, n), as shown below (corresponding lines are colour coded to show transformation):



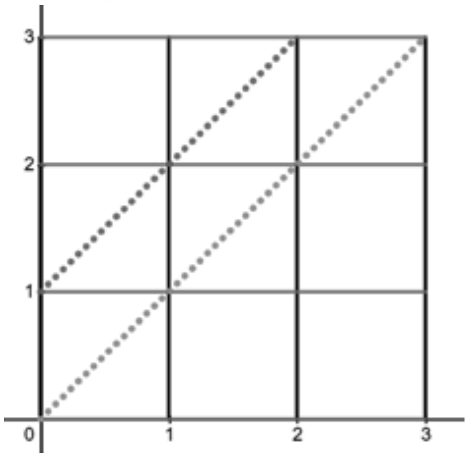
In the above diagram:
$S = \{(1, 0), (0, 1)\}$.

With the starting point at $(0, 0)$ and the ending point at (n, n). Notice that we need not define the boundaries in the positive y-direction, as the height of the figure is directly proportional to the greatest positive x value that can reach the end point.

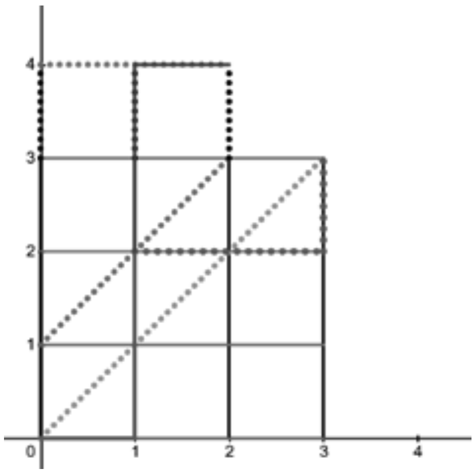
The number of paths from $(0, 0)$ to (n, n) in such a lattice is given by:

$$\frac{1}{n+1}({}^{2n}C_n)$$

This can be obtained by recognising that the current lattice is half that of a normal square lattice from $(0, 0)$ to (n, n).



Note that the number of paths in the triangular lattice above is equivalent to the number of paths from $(0, 0)$ to (n, n) in the square lattice minus the number of paths that touches the upper diagonal (dotted red line) above the diagonal boundary of the triangle lattice (dotted yellow line) when going to (n, n).



The number of paths that needs to be subtracted off can be calculated by flipping the tail of the path after it touches the upper diagonal (dotted red line) for the first time. In the example above, there is a path from $(0, 0)$ to (n, n) in the square lattice (blue line as a whole). However, it touches the upper diagonal.

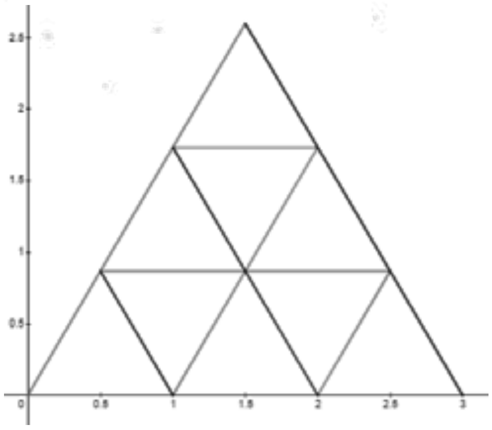
The part after it touches the upper diagonal (dotted blue line) is flipped. All $(0, 1)$ movement is translated into $(1, 0)$ movement and vice versa (dotted blue line becomes purple line). Since the path is flipped at coordinates $(i, i + 1)$ for some non-negative integer i smaller than n, the flipped path will end at $(n - 1, n + 1)$. This can be reasoned out by thinking of the number of upwards and rightwards movements left to be made. Keep in mind that the transformation occurs after the first touch with the upper diagonal, any subsequent touches with the upper diagonal will not signify anything.

Using the transformation, we can map all possible paths from $(0, 0)$ to (n, n) in the square lattice that touches the upper diagonal to all possible paths from $(0, 0)$ to $(n - 1, n + 1)$ in a rectangular lattice. Notice that as each $(1, 0)$ path is mapped to $(0, 1)$ and vice versa, this transformation is one to one. Thus:

$$
\begin{aligned}
{}^{2n}C_n - {}^{2n}C_{n+1} &= \frac{(2n)!}{n!\,n!} - \frac{(2n)!}{(n+1)!\,(n-1)!} \\
&= \frac{(2n)!}{n!\,(n-1)!}\left(\frac{1}{n} - \frac{1}{(n+1)}\right) \\
&= \frac{(2n)!}{n!\,(n-1)!}\left(\frac{1}{n\,(n+1)}\right) \\
&= \frac{(2n)!}{n!\,n!}\left(\frac{1}{(n+1)}\right) \\
&= \frac{1}{(n+1)}\left({}^{2n}C_n\right)
\end{aligned}
$$

F. Case 5, Triangular Lattice, Up (Right), Down (Left), Right

In Case 5, we continue to explore triangular grids as shown below, the aim again being to find the number of paths from $(0, 0)$ to the right-most corner $(n, 0)$ of the figure:



While equilateral triangles look nice and all, S would be defined as
$S = \{(1, 0), (cos\frac{\pi}{3}, sin\frac{\pi}{3}), (cos\frac{\pi}{3}, sin\frac{4\pi}{3})\}$.

Thankfully, we can avoid the messiness of trigonometric functions while solving this lattice. We claim that the formula from $(0, 0)$ to $(n, 0)$ is:

$$\sum_{i=0}^{n} \frac{(2n-i)!}{(n-i+1)! \cdot (n-i)! \cdot i!}$$

First, we observe that the number of moves can range between n to $2n$. This number of moves also describes the number of components that it makes up. For example, in the above triangle lattice, a path that uses 5 moves to reach the end, will contain one horizontal move, two diagonals going up and two diagonals going down. To explain this, a path with $2n$ moves will be made up entirely of diagonal moves, n upwards and n downwards. Because we are going from $(0, 0)$ to $(n, 0)$, every upwards move must be paired with a downwards move such that $j(cos\frac{\pi}{3}, sin\frac{\pi}{3}) + j(cos\frac{\pi}{3}, sin\frac{4\pi}{3}) = (j, 0)$ where $j \leq n$, giving a net upward movement of 0. To arrive at a path with $2n - 1$ moves, a pair of diagonals, one going up and one coming down, will be switched out for a horizontal move. This can be done again and again until all movement consists of horizontal moves, which will take up n moves.

The paths will have i number of horizontal moves, $(n - i)$ number of diagonals going up and $(n - i)$ number of diagonals coming down, for some non-negative integer i between 0 and n (inclusive). Ignoring the horizontal moves, we can use the solution from Case 4 here:

$$\frac{1}{((n-i)+1)}({}^{2(n-i)}C_{n-i})$$

This represents the number of ways to arrange the diagonals as we cannot go down before going up. However, the horizontal moves still exist and we need to take them into account. A path with i number of horizontal moves, $(n - i)$ number of diagonals going up and $(n - i)$ number of diagonals coming down, will have a total of $(2n - i)$ moves. Thus, the number of possible ways to arrange i horizontal moves in a total of $(2n - i)$ moves is ${}^{2n-i}C_i$.

Multiplying them together gives us the total number of possible paths which have i number of horizontal moves, $(n - i)$ number of diagonals going up and $(n - i)$ number of diagonals coming down:

$$\begin{aligned}\frac{1}{(n-i+1)}({}^{2(n-i)}C_{n-i})({}^{2n-i}C_i) &= \frac{1}{(n-i+1)}\left(\frac{(2n-2i)!}{(n-i)!\,(n-i)!}\right)\left(\frac{(2n-i)!}{i!\,(2n-2i)!}\right)\\ &= \frac{1}{(n-i+1)}\left(\frac{(2n-i)!}{(n-i)!\,(n-i)!\,i!}\right)\\ &= \frac{(2n-i)!}{(n-i+1)!\,(n-i)!\,i!}\end{aligned}$$

Summing it up over all possible values of i, we obtain:

$$\sum_{i=0}^{n} \frac{(2n-i)!}{(n-i+1)! \cdot (n-i)! \cdot i!}$$

G. Case 6, 4 Cardinal Directions (North, East, West, South)

Finally, we explore the solutions are a path with $S = \{(1, 0), (0, 1), (-1, 0), (0, -1)\}$, and $R_{n,m}$, for different integer values of m.

The solutions for $m = 0$ and $m = 1$ can be trivially shown to be 1 and 2^n respectively with the formula in Case 3. This is because while all 4 directions are in the set of S, any usage of more than 3 of them will result in the path falling out of the stipulated boundary.

In the case of $m = 1$, any path using the step $(-1, 0)$ will fall out of the boundary. Without loss of generality, let the step $(-1, 0)$ happen in row $m = 0$ and column $n = k$. By definition of a self avoiding path, the only way for that path to reach column $n = k$ in the first place will be from row $m = 1$. However, this leaves no additional space for the path to travel to subsequent columns.

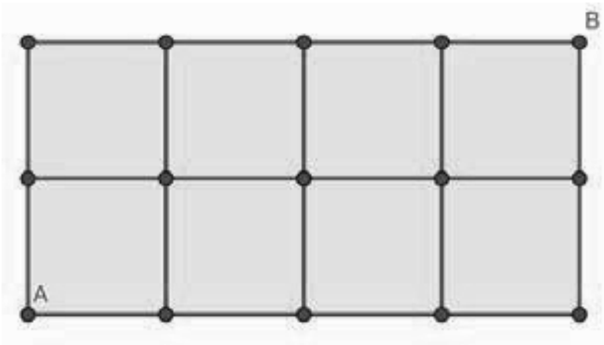
The solutions become more complex past $m = 1$. We provide 2 different solutions to the problem of $m = 2$. To our knowledge, the second solution is completely novel.

i. Solution 1

We claim that the general number of paths is equivalent to

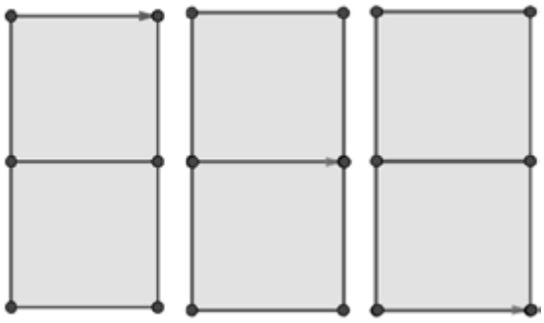
$$f_n = (\frac{1}{2} - \frac{2}{\sqrt{13}})(\frac{3}{2} - \frac{\sqrt{13}}{2})^n + (\frac{1}{2} + \frac{2}{\sqrt{13}})(\frac{3}{2} + \frac{\sqrt{13}}{2})^n + (\frac{i}{2\sqrt{3}})((\frac{1}{2} - \frac{i\sqrt{3}}{2})^n - (\frac{1}{2} + \frac{i\sqrt{3}}{2})^n)$$

In these cases, we try to investigate the number of paths from A to B in a n by 2 grid. The number of paths from the left-bottom corner to the right-top corner in a n by 2 grid will be represented with the f_n. Note that forwards here means to the right and backwards means to the left.



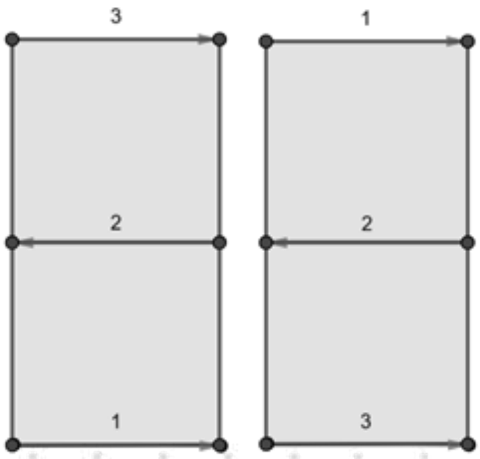
To simplify the grid to find a recurrence relation, we only look at the right-most 1 by 2 rectangle in the n by 2 grid.

Due to the relatively small size of this rectangle, it is quite easy to determine that there are only 5 ways to transverse this rectangle when only considering horizontal movement.



The first three cases all only contain 1 horizontal movement. The number of paths in f_n that contain these horizontal movements in the right-most 1 by 2 rectangle shall be respectively denoted by f_n^0, f_n^2, f_n^3, where n represents the x coordinates of the right-most points. These are just naming conventions inspired by the binary representation of each square. Any notation used will suffice.

The next two cases each contain 3 horizontal movements so the order of which horizontal movement matters.



The first one has the bottom horizontal movement executed first, then the second horizontal movement backwards, lastly with the top horizontal movement forwards. In the second case, the order of the top and bottom movement are switched. Note that no other ways of rearrangement will work, given that they will have to exit the boundary or overlap. These 2 cases are represented respectively by f_n^{1S}, f_n^{12}. Again the notations are arbitrary, in this case the S and 2 are due to how the drawing of horizontal movement in this order will lead to an S and 2 shape. Any other notation will suffice.

Now to prove some basic identities where we will represent f_n^{0}, f_n^{1S}, f_n^{12}, f_n^{2}, f_n^{3} in terms of f_{n-a}^{0}, f_{n-a}^{1S}, f_{n-a}^{12}, f_{n-a}^{2}, f_{n-a}^{3}, where *a* can be any positive integer.

Of course, before that, we have to establish the most basic and intuitive identity:

1) $f_n = f_n^{0} + f_n^{1S} + f_n^{2} + f_n^{3}$

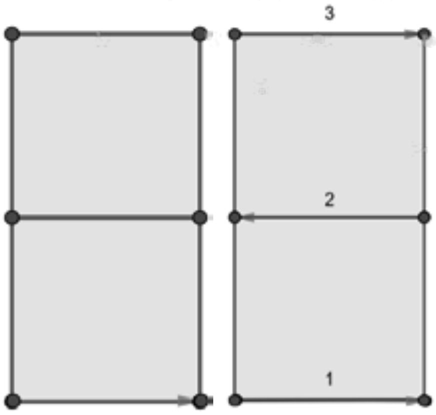
Since f_n^{0}, f_n^{1S}, f_n^{12}, f_n^{2}, f_n^{3} all represent the different ways the horizontal movement can be executed, summing them up will lead to f_n, with the exception of f_n^{12} which ends at the bottom-right corner and is unable to go up without intersecting itself.

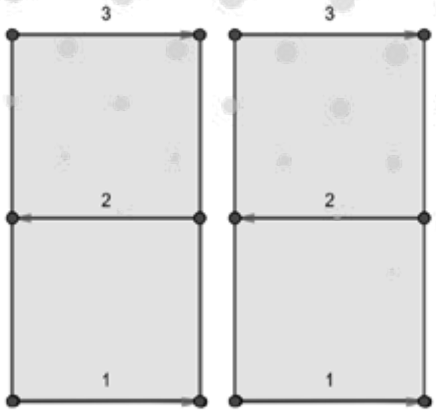
2) $f_n^{0} = f_{n-1}$

This identity is also relatively simple. All paths in a $n - 1$ by 2 grid (f_{n-1}) can be extended as a path in a n by 2 grid with f_n^{0} as the right-most 1 by 2 rectangle, by extending its path with just one horizontal movement forwards.

3) $f_n^{1S} = f_{n-1}^{1S} + f_{n-1}^{3}$

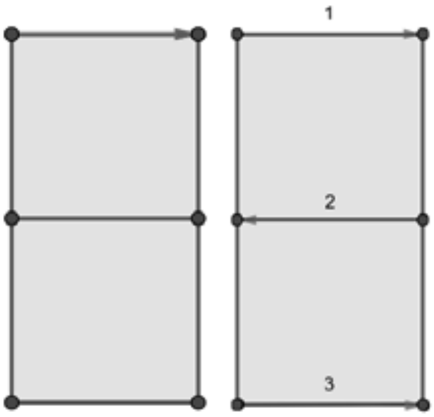
We can see that only in these cases the horizontal movement adds up. The right-most points of the left rectangle and the left-most points of the right rectangle are the same points, just separated for easier recognition of each rectangle.

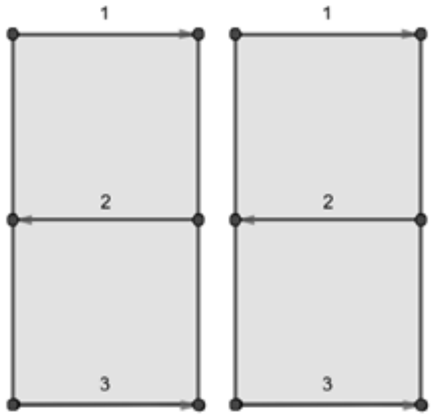




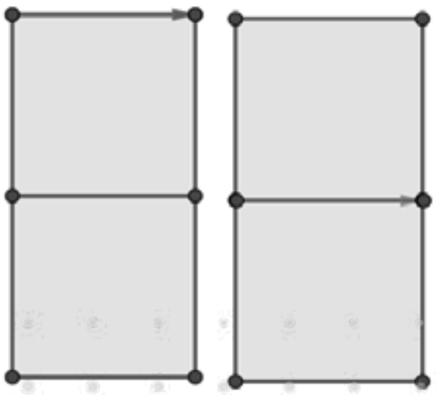
Only horizontal movements were shown as vertical components can and will appear in different locations, as seen here, in the first case, the second upwards vertical movement occurs in the centre while in the second case, it occurs on the left. If the readers are unable to imagine the vertical movements in the following, we would suggest drawing them out on a piece of paper.

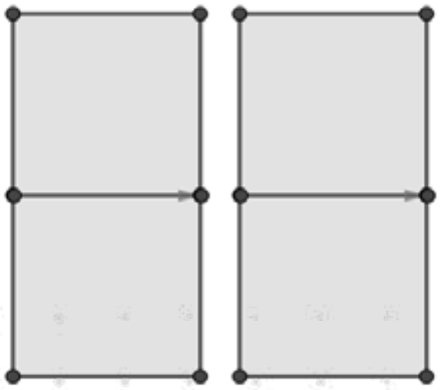
4) $f_{n}^{12} = f_{n-1}^{0} + f_{n-1}^{12}$

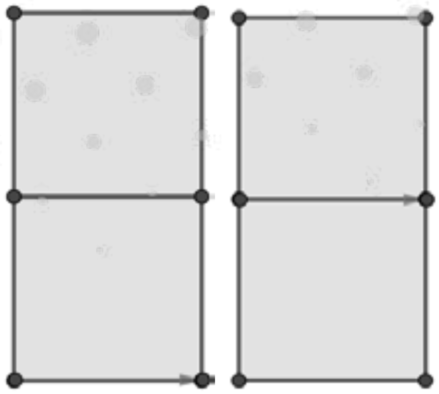




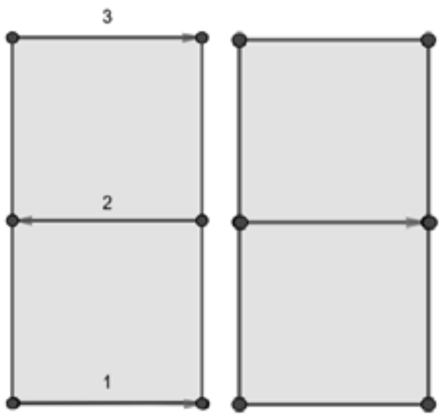
5) $f_{n}^{2} = f_{n-1}^{0} + f_{n-1}^{2} + f_{n-1}^{3}$

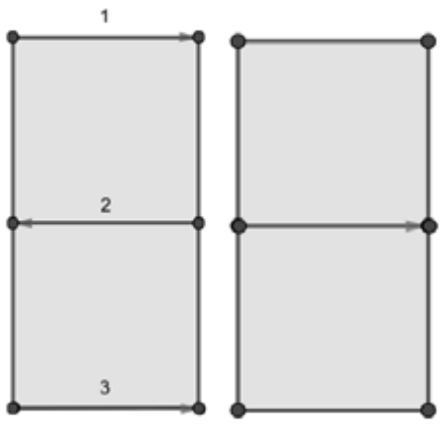




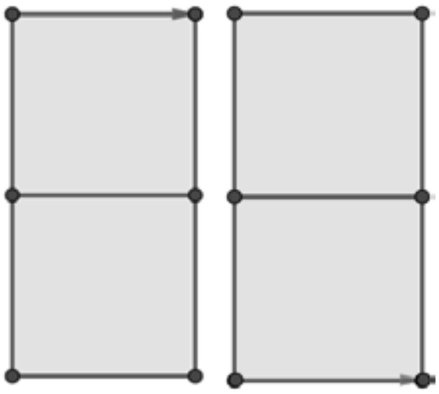


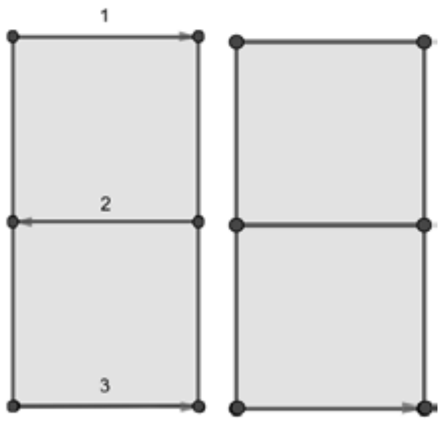
Note that usage of f_n^{1S} , f_n^{12} here will result in overlap as shown in the following cases

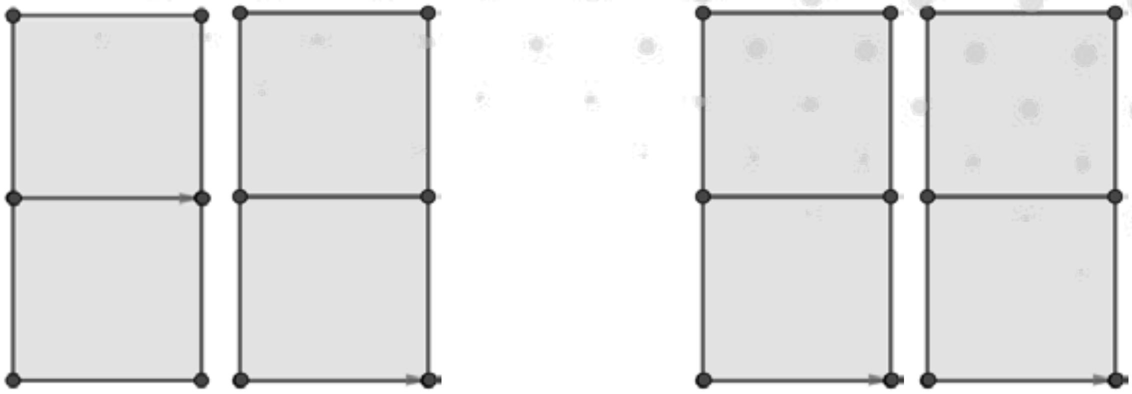




6) $f_n^3 = f_{n-1}^0 + f_{n-1}^{12} + f_{n-1}^2 + f_{n-1}^3$







As a result:

$f_n = f_n^0 + f_n^{1S} + f_n^2 + f_n^3$

$f_n^0 = f_{n-1}$

$f_n^{1S} = f_{n-1}^{1S} + f_{n-1}^3$

$f_n^{12} = f_{n-1}^0 + f_{n-1}^{12}$

$f_n^2 = f_{n-1}^0 + f_{n-1}^2 + f_{n-1}^3$

$f_n^3 = f_{n-1}^0 + f_{n-1}^{12} + f_{n-1}^2 + f_{n-1}^3$

Using the previously derived identities, we thus conclude this recurrence relation

$$\begin{aligned}
f_n &= f_n^0 + f_n^{1S} + f_n^2 + f_n^3 \\
&= f_{n-1} + 2f_{n-1}^0 + f_{n-1}^{1S} + f_{n-1}^{12} + 2f_{n-1}^2 + 3f_{n-1}^3 \\
&= 2f_{n-1} + f_{n-1}^0 + f_{n-1}^{12} + f_{n-1}^2 + 2f_{n-1}^3 \\
&= 2f_{n-1} + f_{n-1}^0 + f_{n-1}^{12} + f_{n-1}^2 + 2f_{n-1}^3 + (2f_{n-1} - 4f_{n-2} - 2f_{n-2}^0 - 2f_{n-2}^{12} - 2f_{n-2}^2 - 4f_{n-2}^3) \\
&= 4f_{n-1} - 3f_{n-2} + f_{n-1}^{12} + f_{n-1}^2 + 2f_{n-1}^3 - 2f_{n-2}^0 - 2f_{n-2}^{12} - 2f_{n-2}^2 - 4f_{n-2}^3 \\
&= 4f_{n-1} - 3f_{n-2} + 2f_{n-2}^0 + f_{n-2}^{12} + f_{n-2}^2 - f_{n-2}^3 \\
&= 4f_{n-1} - 3f_{n-2} + 2f_{n-3} + f_{n-3}^0 \\
&= 4f_{n-1} - 3f_{n-2} + 2f_{n-3} + f_{n-4}
\end{aligned}$$

Where the factor added in the fourth row is equal to zero, and simply helps in simplification. To solve this using generating functions would be long, we thus utilise characteristic polynomials instead.

We guess a solution $f_n = r^n$, where r is constant. Doing so gives us

$r^n = 4r^{n-1} - 3r^{n-2} + 2r^{n-3} + r^{n-4}$

$r^4 - 4r^3 + 3r^2 - 2r - 1 = 0$

Which can be factored as $(r^2 - r + 1)(r^2 - 3r - 1)$, which gives the roots

$r = \frac{3}{2} - \frac{\sqrt{13}}{2}, \frac{3}{2} + \frac{\sqrt{13}}{2}, \frac{1}{2} + \frac{i\sqrt{3}}{2}, \frac{1}{2} - \frac{i\sqrt{3}}{2}$

This means that the general formula can be written as

$f_n = c_1(\frac{3}{2} - \frac{\sqrt{13}}{2})^n + c_2(\frac{3}{2} + \frac{\sqrt{13}}{2})^n + c_3(\frac{1}{2} + \frac{i\sqrt{3}}{2})^n + c_4(\frac{1}{2} - \frac{i\sqrt{3}}{2})^n$

To solve for the constants, we need to look at our initial conditions.

$f_0 = 1$ is clear, since to get from $(0, 0)$ to $(2, 0)$ one can only move right.

$f_1 = 4$ is also clear, since this is simply a special case of $n = 1$.

$f_2 = 12$ as well as $f_3 = 38$ is slightly more difficult. We found these using the python code found in the Appendix.

We have to solve the following recurrence relation.

$c_1 + c_2 + c_3 + c_4 = 1$

$c_1(\frac{3}{2} - \frac{\sqrt{13}}{2}) + c_2(\frac{3}{2} + \frac{\sqrt{13}}{2}) + c_3(\frac{1}{2} + \frac{i\sqrt{3}}{2}) + c_4(\frac{1}{2} - \frac{i\sqrt{3}}{2}) = 4$

$c_1(\frac{3}{2} - \frac{\sqrt{13}}{2})^2 + c_2(\frac{3}{2} + \frac{\sqrt{13}}{2})^2 + c_3(\frac{1}{2} + \frac{i\sqrt{3}}{2})^2 + c_4(\frac{1}{2} - \frac{i\sqrt{3}}{2})^2 = 12$

$c_1(\frac{3}{2} - \frac{\sqrt{13}}{2})^3 + c_2(\frac{3}{2} + \frac{\sqrt{13}}{2})^3 + c_3(\frac{1}{2} + \frac{i\sqrt{3}}{2})^3 + c_4(\frac{1}{2} - \frac{i\sqrt{3}}{2})^3 = 38$

This system can be easily solved via Gaussian Elimination, albeit tediously. Doing so gives

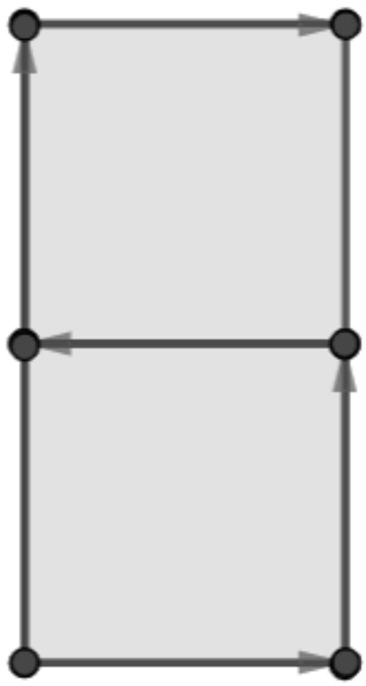
$c_1 = \frac{1}{2} - \frac{2}{\sqrt{13}}, c_2 = \frac{1}{2} + \frac{2}{\sqrt{13}}, c_3 = -\frac{i}{2\sqrt{3}}, c_4 = \frac{i}{2\sqrt{3}}$

$f_n = (\frac{1}{2} - \frac{2}{\sqrt{13}})(\frac{3}{2} - \frac{\sqrt{13}}{2})^n + (\frac{1}{2} + \frac{2}{\sqrt{13}})(\frac{3}{2} + \frac{\sqrt{13}}{2})^n + (-\frac{i}{2\sqrt{3}})(\frac{1}{2} + \frac{i\sqrt{3}}{2})^n + (\frac{i}{2\sqrt{3}})(\frac{1}{2} - \frac{i\sqrt{3}}{2})^n$

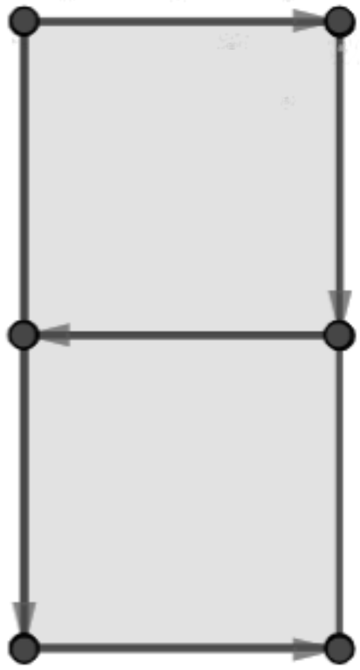
$= (\frac{1}{2} - \frac{2}{\sqrt{13}})(\frac{3}{2} - \frac{\sqrt{13}}{2})^n + (\frac{1}{2} + \frac{2}{\sqrt{13}})(\frac{3}{2} + \frac{\sqrt{13}}{2})^n + (\frac{i}{2\sqrt{3}})((\frac{1}{2} - \frac{i\sqrt{3}}{2})^n - (\frac{1}{2} + \frac{i\sqrt{3}}{2})^n)$

ii. Solution 2

Our second solution comes as a product of us noticing that there are only 2 types of paths that include a left motion in the $(n, 2)$ grid.

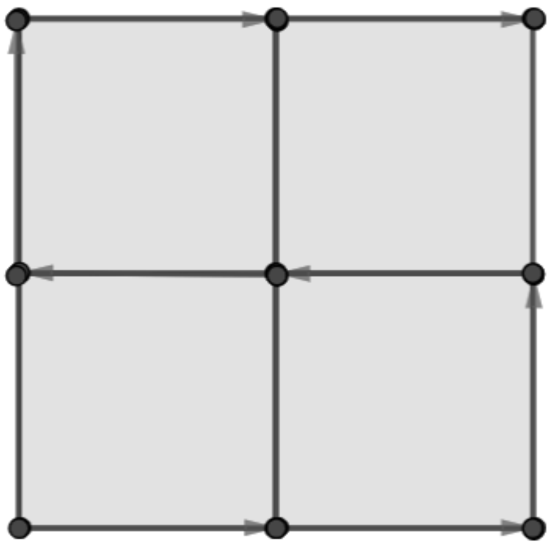


(***B***)



(***C***)

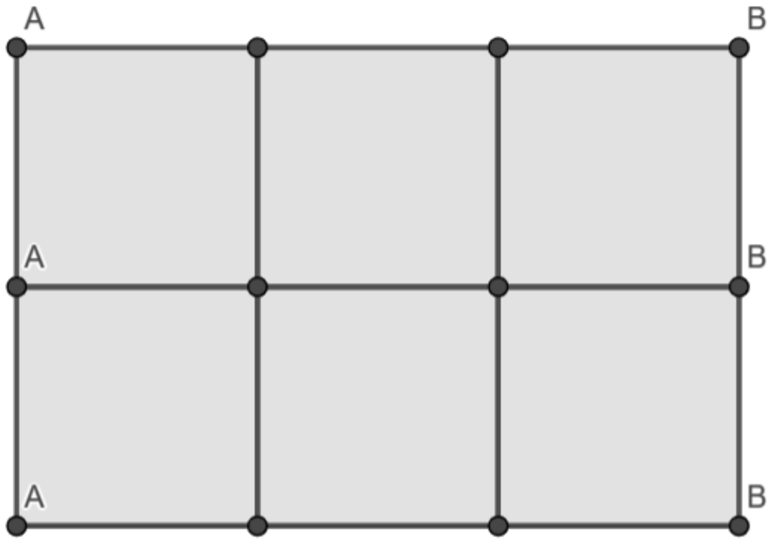
Both these examples shown take up the entire vertical height of the grid. Of course, the length of the these examples can be increased:



(***C*** with a length of 2)

But for a certain length k, there are only 2 unique paths that contain k number of left movements. We represent the first case on the left with the letter ***B***, and the second, on the right, with the letter ***C***.

With four directions to possibly move in, we must also account for vertical movement. We already proved that any left movement will have to result in either a ***B*** or a ***C***, so the rest of the moves can only take into account upwards, downwards and right movements. Luckily for us, this means we can use a milder version of Case 3's solution to solve this problem, as this grid problem is essentially Case 3's, just without the diagonals and is tilted sideways.



The number of solutions of how many paths there are without left motion from any point A to any point B is given by 3^k where k is the distance between point A and point B, which is 3 units in this example. This is intuitive as there are only 3 possible ways to go right, and the only allowed moves are up, down and right. For a more rigorous proof, a similar proof to Case 3 can be applied, just without the diagonal movement. We refer to such a path using the letter ***A***.

This means that all paths from the origin to the point $(n, 2)$ in an n by 2 grid that can transverse all up, down, left and right movements can be expressed as a chain of ***A***, ***B*** and ***C***s.

We then proceed to figure out some rules that the chain of ***A***, ***B***, ***C***s must obey. These are the rules and why they must be true for such a path to exist as a solution to the problem.

1) The chain must start with either ***A*** or ***B***, but not ***C***.

This is easy to verify as only ***A*** and ***B*** can start on the bottom-left corner. ***A*** can start from any point on the left column, which includes the bottom-left corner, which is the starting point for the solution. ***C***, on the other hand, starts from the top-left corner and thus cannot be the start of the solution.

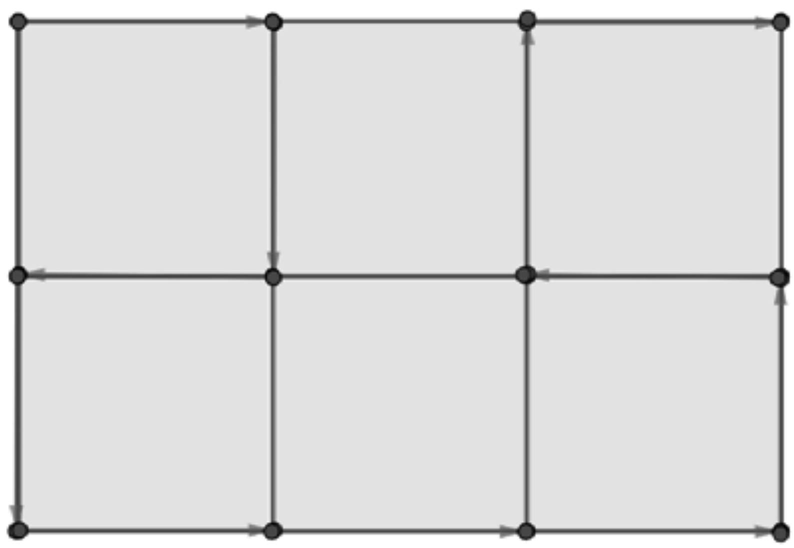
2) The chain must end with ***A*** or ***B***, but not ***C***.

Again this is easy to verify, as similar to above, only ***A*** and ***B*** can end on the top-right corner, whereas ***C*** ends on the bottom-right corner.

3) The letters cannot be adjacent to themselves.

To verify this we disprove three scenarios, that ***AA***, ***BB*** and ***CC*** all cannot exist in a possible solution. Since ***AA*** is just one big ***A***, and thus will be double counting if we included both possibilities. Since ***A*** can start from any point on the bottom and end at any point on the top, putting two ***A***s together will just result in a larger ***A***. ***BB*** is easy to disprove as it starts on the bottom-left and ends on the top-right. This means that anything after ***B*** has to start from the top-left, which happens to be ***A*** and ***C***, ***A*** because it can start from any point from the left column. ***CC*** does not work by the same logic as ***BB***, just in the reversed direction, as ***C*** starts from the bottom-left and ends on the top-right.

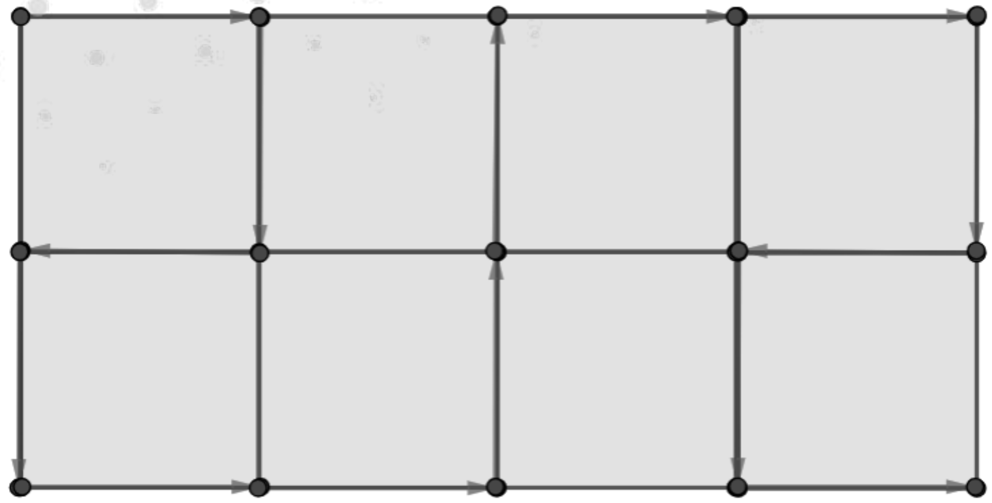
4) A margin of 1 must be inserted between 2 letters.



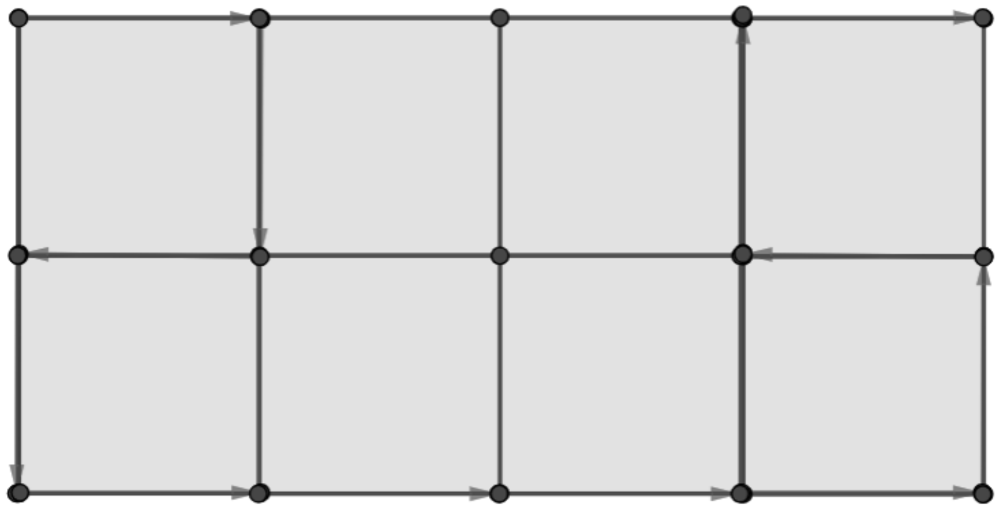
The diagram above shows ***CB***, with a margin in the middle. This is required to prevent overlap, and since all ***A***, ***B*** and ***C***s all start from the corner, they include all other vertical points at the starting horizontal column in their solution, meaning that all ***A***, ***B*** and ***C***s require a new column to start upon.

5) ***B*** and ***C*** both have a minimum length of 1, whereas ***A*** has a minimum length of 0.

The reason for ***B*** and ***C***'s minimum is simple, as without a tangible length, there will be no left movement, and at that point ***B*** and ***C*** are just vertical movement, which is ***A***. ***A*** can have a minimum length of 0 so as to account for a straight vertical line, or just a point between 2 margins. The example on the top is ***CAC***, with ***A*** being just a straight line with a length of 0; the example at the bottom is ***CAB***, with ***A*** just being a point with length 0.



(***CAC***)

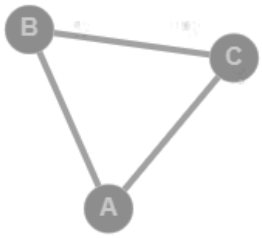


(***CAB***)

Thus having these 5 rules, we can now set out to find the solutions using a combination of ***A***, ***B***, ***C***s. Firstly, we have to find out given a certain number of ***A***, ***B*** and ***C***s in a solution, how many possible ways of arranging them exists, while ensuring that the first 3 rules are not violated.

We start by finding the number of ways to sort different numbers of path segments ***A***s, ***B***s, and ***C***s, in accordance with the first 3 restrictions: the chain must start and end with ***A*** or ***B***, and path segments of the same alphabet cannot be adjacent to each other.

Solving with combinatorics has proven to be ineffective. Generally, brute forcing such a problem leads to too many cases to consider. Instead, we approach this via a graph, as shown below.



This graph shows us what pathways we can take in a certain number of movements, starting from any node, and what nodes we cross along the way. Note that the paths do not directly lead back to itself (ie. we cannot return to the starting node in exactly 1 movement). This represents the third criteria of our chain: letters cannot be adjacent to themselves. In context to our problem, if we start on ***A*** as our zeroth position, we are able to reach ***B*** or ***C*** after one movement, and ***A***/***C*** or ***A***/***B*** after the second movement respectively. Restricting the total number of times we cross each node, we can find how many ways are there that allow us to start from ***A*** or ***B***, and end at ***A*** or **B**.

First, note that crossing each node is equivalent to taking the path to each node. In other words, crossing to node ***A*** is the same as taking a path from ***B*** to ***A*** or from ***C*** to ***A***. Note this is true as ***A*** can only be reached by these 2 paths, and that these 2 paths will only reach ***A***. In order to distinctively relate each path to the node it leads to, we can thus assign arbitrary weights to each of the paths. In this case, we assign x to both path leading to ***A***, y to both paths leading to ***B***, and z to both paths leading to ***C***.

From this, we can form the adjacency matrix as below.

$$\begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}$$

This matrix represents the above graph. The first row states that starting from ***A***, we can either take a path with weigh y to ***B*** or weight z to ***C***. Similarly, the second and third row map out the first movements from starting points ***B*** and ***C*** respectively. As mentioned in the graph above, a node starting at ***A*** cannot take a path to itself; thus the weight of such a movement is 0.

For a total of $a + b + c$ movements, we simply raise the matrix to that exponent.This maps out for us all possible movements for all possible starting positions and all possible crossings of nodes ***A***, ***B*** and ***C***. With matrix multiplication being commutative for exponent powers, we can get the following:

$$\begin{bmatrix}0 & y & z\\ x & 0 & z\\ x & y & 0\end{bmatrix}^{a+b+c} = \begin{bmatrix}0 & y & z\\ x & 0 & z\\ x & y & 0\end{bmatrix}\begin{bmatrix}0 & y & z\\ x & 0 & z\\ x & y & 0\end{bmatrix}^{a+b+c-1} = \begin{bmatrix}0 & y & z\\ x & 0 & z\\ x & y & 0\end{bmatrix}^{a+b+c-1}\begin{bmatrix}0 & y & z\\ x & 0 & z\\ x & y & 0\end{bmatrix}$$

Thus, the row of the final matrix tells us which node we start from (or rather, what we cannot start from). This can be seen in the second expression in the equality. Each weight in the rows of the resulting matrix is contributed by that same row in the first matrix. Similarly, the column of the resulting matrix tells us what node we end on. In the third expression in the inequality, the columns of the last matrix multiplication make up the terms in the corresponding columns of the resulting matrix. For example, the top row tells us we start on ***B*** or ***C*** (we cannot start from an ***A*** as the weight to ***A*** is 0), and the left most column tells us that we must end on ***A***. Thus, the top left column must start on either ***B*** or ***C***, and end on ***A***. Note that we need not control what nodes are crossed in between the start and the end, as by definition one node cannot lead to itself, fulfilling our third criterion.

We can then restrict the number of times we cross a node by finding the coefficient of certain terms in certain entries in the obtained matrix. Each time we "cross" a "path", we "pick up" the weight of the paths. For example, crossing to A and then to B gives us a weight of xy. Thus, given a ***A***s, b ***B***s, and c ***C***s, we simply need to find the coefficient of $x^a y^b z^c$. Note that $a + b + c$ movements will give us that same number of total weights. This is true as $a + b + c$ movements must cross $a + b + c$ paths, giving us $a + b + c$ weights.

Lastly, we need to restrict the starting and ending points. Note that the first path segment must be an ***A*** or ***B***, and having these two choices means that it must follow a ***C***. Another way to view this is that only if the zeroth position is a ***C*** can we have the choice of ***A*** or ***B*** in the first position. In other words, we are only concerned about paths away from ***C***, with weights x to ***A*** and y to ***B***. Thus, we are concerned about the entries in the last row. In addition, for the final entry not to be a ***C***, the final weight picked up cannot be a z. Essentially, we ignore all paths found in the right column of the final matrix, as they represent picking up z as the final weight. Thus, we are only concerned with paths mapped in the first and second cell of the third row.

As stated above, from all the paths mapped into the first and second cell of the third row, we search for the coefficient of the $x^a y^b z^c$ term. In notation, it looks like this:

$$[x^a y^b z^c]\left(\begin{bmatrix}0 & y & z\\ x & 0 & z\\ x & y & 0\end{bmatrix}^{a+b+c}_{3,1} + \begin{bmatrix}0 & y & z\\ x & 0 & z\\ x & y & 0\end{bmatrix}^{a+b+c}_{3,2}\right) \quad \text{------(1)}$$

Where $[k]$ finds the coefficient of some k. With the formula above, we have found the number of ways to arrange a ***A***s, b ***B***s, and c ***C***s such that they fulfil the 3 criteria established.

Now, we have to find the number of unique paths that can be derived from a specific chain of ***A***s, ***B***s and ***C***s while obeying rules 4 and 5. For example, we have a chain ***ACA*** that has a length of 4. Since between every letter is a margin, the combined lengths of the 2***A***s and ***C*** must be $4 - 2 = 2$. Since the minimum length for ***C*** is 1, and the minimum length of ***A*** is 0, in this example, ***C*** can have a length of 1 or 2.

Recall from earlier in this segment that for a certain length ***C*** represents only 1 path, while ***A*** represents 3^k where k is the length of ***A***. For the scenario where ***C***'s length is 1, either the front or back ***A*** could be the one which has length 1, so the total number of unique paths is $2(1)(3^1)(3^0) = 6$. For the scenario where ***C*** has a length of 2, both As have lengths of 0, so the number of unique paths is $(1)(3^0)(3^0) = 1$. In the end, the number of unique paths that can be derived from ***ACA*** with a length of 4 is 7.

In order to generalise such a formula for any possible chains, we first have to separate the solutions based on the number of ***A***s that have a length 0. The lower bound for the number of As that have a length of 0 is simple to calculate: $2a + 2b + 2c - n - 1$, where a, b and c represent the number of ***A***, ***B*** and ***C***s respectively. This is derived by first subtracting the margins $a + b + c - 1$ from n. Note that as we are finding the minimum number of ***A***s that have a length of 0, we need the maximum number of As with a length greater than 0. In other words, the lengths of ***B*** and ***C*** should be their minimum. Since ***B*** and ***C*** have a minimum length of 1, we subtract the minimum length of $b + c$. This number thus now represents the remaining length not taken up by ***B***s or ***C***s. In other words, the number of ***A***s that do not have a length of 0. To get the number of ***A***s with a length of 0, this number is subtracted from a. This gives us the expression above. Of course, this value has cannot be lower than 0 (ie. cannot have a negative number of ***A***s) so we obtain the lower bound $max(2a + 2b + 2c - n - 1, 0)$.

The maximum value of ***A***s that can have a length of 0 is obviously a. Also note that different ***A***s can have the height of 0, so the summation needs to be multiplied by: ${}^{a}C_{a_0}$, where a_0 represents the number of ***A***s that have height 0. As a result the final summation will look something like: $\sum_{a_0 = max(2a+2b+2c-n-1, 0)}^{a} ({}^{a}C_{a_0})(f(a, a_0, b, c, n))$.

$f(a, a_0, b, c, n)$ represents the number of unique paths that share the same chain of ***A***s, ***B***s and ***C***s (e.g. ***ACA***), given the specific number of ***A***s that have length 0. To do this, we can create a string of numbers, each representing the length of a certain ***A***, ***B*** or ***C*** in the combination. This string of numbers add up to $p = n - b - c - a + 1$, the total height of all ***A***s, ***B***s and ***C***s combined. Take note that all ***A***s with length 0 are ignored as they only have $3^0 = 1$ unique path each and their heights are fixed. This means that we will just multiply our number of paths by 1. However, we cannot ignore ***B***s and ***C***s as any change in their length will affect the lengths of other letters.

This is then cycled through all possible combinations, of which order matters. A formula for this will be proved later.

For example, for 4 letters and $p = 6$, all combinations will be: {1113, 1131, 1311, 3111, 1122, 1212, 2112, 1221, 2121, 2211}. As these are only for finding the number of unique paths within a certain chain, rules 1, 2, 3, can be ignored; so rearranging the positions of certain letters for visualisation is possible. For example, the above numbers represent the chain ***ACBA*** where no ***A***s are of 0 height. If we change the chain to ***AABC***, all the numbers in the fourth position swap with the numbers in the second position, but the overall strings of numbers remain unchanged (just with numbers in different positions for simplification).

After swapping all the ***A***s to the front (the number sequences remain unchanged), we can split the ***A***s from the ***B***s and ***C***s in the string into 2 parts: {13|11, 11|31, 11|13, 31|11, 12|21, 12|12, 22|11, 11|22, 21|21, 21|12}. Looking closely we can categorise the solutions. {13|11, 22|11, 31|11} has the total length of all the ***A***s as 4. {12|12, 12|21, 21|12, 21|21} has the length as 3, and lastly, {11|13, 11|22, 11|31} has the length as 2. Note that all such strings are unique, as we only change the positions between the 2nd and the 4nd numbers and begin with a unique string.

This can be proven by contradiction. Suppose two unique strings exist that, with certain number positions swapped, will form the same string. This implies that every number not swapped must be the same in both strings, and that numbers swapped must be the same. We have a contradiction: for all numbers swapped to be the same implies that all numbers in the original two strings are the same. As all strings are unique, all strings modified as above are unique.

The lowest value that the total length of ***A***s can get is given by $(a - a_0)$, since the ***A***s that do not have a length of 0 have a minimum length of 1. The maximum value is given by $p - b - c$. So the summation will be: $\sum_{a_t = a-a_0}^{p-b-c}$, where a_t represents the total length of the ***A***s. Inside the summation, we have to multiply the number of possible sequences of the first half (only ***A***s), with that of the second half (***B***s and ***C***s). Within the first half of the string that only has ***A***'s, this can be done by noticing that for a particular half-string, we can spread it out into 1s, for the string 1122|1131, the first half is 1122, and then this is converted into $\underline{1}$, $\underline{1}$, 1, $\underline{1}$, 1, $\underline{1}$, where underlined 1s add up with every non-underlined 1 before it, so long as there is no other underlined 1 between them. For the half-string 31, the expanded form is 1, 1, $\underline{1}$, $\underline{1}$. Notice that the number of 1s is equals to a_t, and the number of underlined 1s is equal to $(a - a_0)$. Since the last 1 always has to be underlined (since it has to be the end), the number of ways to rearrange this expanded form, and hence sum it up differently, is ${}^{a_t - 1}C_{a-a_0-1}$. The same can be done to the second half, which represents ***B***s and ***C***s. However, in the second case, the number of 1s is $(p - a_t)$, while the number of underlined 1s is $b + c$. Thus the expanded form for the second half is ${}^{p- a_t - 1}C_{b+c-1}$.

We also cannot forget to multiply this by 3^{a_t} , as for each ***A*** with length k, it has 3^k number of unique paths in it. And because all the different ks add up to a_t , multiplying the different 3^k s together will just give us 3^{a_t} by law of indices.

Looking at $f(a, a_0, b, c, n)$ as a whole, we now get : $\sum_{a_t = a-a_0}^{p-b-c} ({}^{a_t - 1}C_{a-a_0-1})({}^{p-a_t-1}C_{b+c-1})$. What may seem alright at a first glance is quickly disproved when taking $\sum_{a_0 = max(2a+2b+2c-n-1, 0)}^{a}$ into account. We now need to account for the cases where this formula gives undefined values.

If $a_0 = a$, and $a_t = a - a_0$, then ${}^{a_t - 1}C_{a-a_0-1}$ will become ${}^{(-1)}C_{(-1)}$ which is undefined. Thus, what we did was simply remove $a_0 = a$ and add it back later, since it was a much simpler case and allowed for some cancelling so as to prevent ${}^{(-1)}C_{(-1)}$ from happening. Thus our $\sum_{a_0 = max(2a+2b+2c-n-1, 0)}^{a}$ becomes $\sum_{a_0 = max(2a+2b+2c-n-1, 0)}^{a-1}$. Outside the summation that we changed, we then added a calculation for if all ***A***s were of 0 length. This is just ${}^{p-a_t-1}C_{b+c-1}$, where $a_t = 0$, giving ${}^{p-1}C_{b+c-1}$. No additional multipliers are needed as for all ***B***s and ***C***s of a certain length, they each only give 1 unique path. Thus as a result the final $f(a, a_0, b, c, n)$ is

$$\sum_{a_0 = max(2a+2b+2c-n-1, 0)}^{a-1} (({}^{a}C_{a_0}) \sum_{a_t = a-a_0}^{n-2b-2c-a+1} (({}^{a_t - 1}C_{a-a_0-1})({}^{n-b-c-a-a_t}C_{b+c-1})(3^{a_t}))) + {}^{n-b-c-a}C_{b+c-1}. \text{ ---(2)}$$

The equation shown already has $p = n - b - c - a + 1$ substituted into it. Now, there is a second issue with the formula, and that happens when a is 1 and b and c are both 0. If $a = 1$, then a_t has a maximum value of n. This means that if $a_t = n$, ${}^{n-b-c-a-a_t}C_{b+c-1} = {}^{(-1)}C_{-(1)}$, which cannot be accepted. However, since having a as 1 and both b and c as 0, there is definitely only 1 way to arrange it, and the number of unique paths is just 3^n, as there are no margins, it can be put outside the summation and added at the end. As this means that we cannot account for $b = 0$ and $c = 0$ in the summation, we will have to edit the lower bounds as shown later on

Now, all that is left is to define 3 summations, which gives us all possible combinations of ***A***s, ***B***s and ***C***s. They will be at the start of the formula, in the form of $\sum_{c=l_c}^{u_c} \sum_{b=l_b}^{u_b} \sum_{a=l_a}^{u_a}$, where u_c, u_b, u_a, l_c, l_b, l_a are all upper and lower bounds to be found. This will sum up all possible combinations of ***A***s, ***B***s and ***C***s.

$l_c = 0$ as the smallest number of ***C*** you can have in a chain is 0. $u_c = floor(\frac{n}{3})$ as the most number of ***C***s that can exist exists in the chain ***ACACACA***… , where it ends with ***A***, and ***A***s all have a length of 0. This means that for each ***C***, they need to account for the minimum length of 1 they have, and the 2 margins next to it. We then need to floor $(\frac{n}{3})$ as it must round downwards as we cannot have a fraction of a letter.

$l_b = 1 - ceil(\frac{3c}{n})$. l_b can start at 0 as well, but since having both b and c equal 0 will break the formula (1), $l_b = 1 - ceil(\frac{3c}{n})$ ensures that b will start from 0 if $c \neq 0$, but will start from 1 if $c = 0$. $\frac{3c}{n}$ could be $\frac{Vc}{n}$, where V is just a constant between 0 to 3, 3 was just chosen because the upper bound for c is $floor(\frac{n}{3})$.$u_b = min(n - 3c, floor(\frac{n-c+2}{3}))$ is a little harder to derive as u_2 as c could be so large that some positive a must be used due to ***A***'s ability to have a minimum length of 0. In these cases, $u_b = n - 3c$ (n minus the minimum total length of ***C*** and their adjacent margins). For example: we have a chain ***ACACACACA*** that has length 12, we can see that there is no way to place a ***B*** inside, as $12 - 8 - 4 = 0$, with 8 margins of 1 and 4 ***C***s with a length of 1. If the chain had a length of 13 instead, 1 ***B*** can replace an ***A*** such as in the case of ***ACACACACB***, as ***B*** has a minimum length of 1, while ***A*** has a minimum length of 0, we get a difference in length of $(1 - 0)$, so $\frac{13-12}{1-0} = 13 - 12 = 1$. However, if c is small and ***C***s can be surrounded by ***B***'s, then $u_b = floor(\frac{n-c+2}{3})$ as the chain will be ***BCBCBAB***… , ending in either an ***A*** or a ***B***. By taking $n - c$, we change the chain to ***BABABAB***… , and since the first and last terms do not have margins before and after them respectively, we add 2 and divide by 3 (again, accounting for margins). Now every ***B*** (including the start and end since we added 2) has 2 margins next to them and a minimum length of 1. We then round downwards as you cannot have a fraction of a letter, and take the lesser of the two.

$l_a = |c - b + 1|$, as the chain ***BCBCBCB*** will not need any ***A*** whatsoever, but extra ***C***s, such as ***BCBCBCBCACA*** (4 ***B***s and 5 ***C***s) and extra ***B***s such as ***BCBCBCBABAB*** (6 ***B***s and 3 ***C***s) will need ***A***s to be present to make sure that rules 1 and 2 are obeyed.
$u_a = min(n - 2b - 2c + 1, c + b + 1)$, as a is limited by

1) the amount of space left. Since every ***B*** and ***C*** has a minimum length of 1 and has a margin to the right except for the last ***B***, $n - 2b - 2c + 1$ gives the maximum additional space left,

which directly translates to how many ***A***s we can put into the chain since ***A***s can have a minimum length of 0 and only the margin to the right needs to be considered

2) by the number of b and c, as the chain ***ABABACABA*** is unable to fit anymore ***A***s inside as then it would violate rule 3, thus $c + b + 1$.

As a result we also take the lowest value here.

Thus, the full formula will look like:

$$\left(\sum_{c=0}^{floor(\frac{n}{3})} \sum_{b=1-ceil(\frac{3c}{n})}^{min(n-3c, floor(\frac{n-c+2}{3}))} \sum_{a=|c-b+1|}^{min(n-2b-2c+1, c+b+1)} \left([x^a y^b z^c] \left(\begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}^{a+b+c}_{3,1} + \begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}^{a+b+c}_{3,2} \right) \times \right.\right.$$

$$\left(\sum_{a_0 = max(2a+2b+2c-n-1, 0)}^{a-1} \left(\left({}^{a}C_{a_0} \right) \sum_{a_t = a-a_0}^{n-2b-2c-a+1} \left(\left({}^{a_t - 1}C_{a-a_0-1} \right) \left({}^{n-b-c-a-a_t}C_{b+c-1} \right) \left(3^{a_t} \right) \right) \right) + {}^{n-b-c-a}C_{b+c-1} \right) \Bigg) \Bigg) +$$

In summary,

We first take the number of paths for one single ***A*** (3^n).

We then add the number of paths that have at least 1 ***B***/***C.***

To find this term, we take a sequence of a ***A***'s, b ***B***'s, and c ***C***'s (***AAAA***…***BBBB***…***CCCC***), not satisfying the rules yet (we will take this into account in the next paragraph). We generate all the ways to split the height of the rectangle to all individual a ***A***'s, b ***B***'s and c ***C***'s. To find the total number of paths from that sequence, we then take into account the number of paths each ***A***, ***B*** and ***C*** contribute, which is 3^k for ***A***, 1 for ***B***, and 1 for ***C***. (2) then gives the total number of paths of form ***AAAA***…***BBBB***…***CCCC***

Then, we need to rearrange these ***A***, ***B*** and ***C*** in a way satisfying our conditions, given by the matrix term (1). Since each rearrangement would give the same number of paths, we can multiply it directly. Finally, we vary the number of ***A***, ***B*** and ***C*** to get our final sum.

The values of $n = 0, 1, 2, 3, 4, 5, 6$, will get you the values $1,\ 4,\ 12,\ 38,\ 125,\ 414,\ 1369$. Which is the correct values for the sequence.

As a demonstration, take $n = 7$:

First, let us find the values for the summations. Subbing in values gives: u_c, u_b, u_a, l_c, l_b, l_a

$l_c = 0$

$u_c = floor(\frac{7}{3}) = 2$

$l_b = 1 - ceil(\frac{3c}{7})$

$u_b = min(7 - 3c,\ floor(\frac{9-c}{3}))$

$l_a = |c - b + 1|$

$u_a = min(8 - 2b - 2c, c + b + 1)$

So the summations will look like: $\sum_{c=0}^{2} \sum_{b=1-ceil(\frac{3c}{7})}^{min(7-3c, floor(\frac{9-c}{3}))} \sum_{a=|c-b+1|}^{min(8-2b-2c, c+b+1)}$

Our c can be equals to 0, 1, 2:

- When c is 0, b can range from 1 to 3:
  - When $b = 1$, a can range from 0 to 2.
  - When $b = 2$, a can range from 1 to 3.
  - When $b = 3$, a can only be 2.
- When c is 1, b can range from 0 to 2:
  - When $b = 0$, a can only be 2.
  - When $b = 1$, a can range from 1 to 3.
  - When $b = 2$, a can range from 0 to 2.
- When c is 2, b can range from 0 to 1
  - When $b = 0$, a can only be 3
  - When $b = 1$, a can only be 2

This means that the possible combinations of letters for $n = 7$ are as follows:
B, ***AB, AAB***, ***ABB***, ***AABB***, ***AAABB***, ***AABBB***, ***AAC***, ***ABC***, ***AABC***, ***AAABC***, ***BBC***, ***ABBC***, ***AABBC***, ***AAACC***, ***AABCC***.

Below are the matrices we need (note that given a ***A***s, b ***B***s, and c ***C***s, we simply need to find the coefficient of $x^a y^b z^c$):

$$\sum_{i=1}^{2} \begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}_{3,i} = x + y$$

$$\sum_{i=1}^{2} \begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}^2_{3,i} = 2xy$$

$$\sum_{i=1}^{2} \begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}^3_{3,i}$$

$$= x^2y + x^2z + xy^2 + 2xyz + y^2z$$

$$\sum_{i=1}^{2} \begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}^4_{3,i} = 2x^2y^2 + 4x^2yz + 4xy^2z$$

$$\sum_{i=1}^{2} \begin{bmatrix} 0 & y & z \\ x & 0 & z \\ x & y & 0 \end{bmatrix}^5_{3,i} = x^3y^2 + x^3z^2 + x^2y^3 + y^3z^2 + 3xy^2z^2 + 3x^2yz^2 + 2x^3yz + 8x^2y^2z + 2xy^3z$$

Note that we do not need to take the matrix to an exponent greater than 5 as we can fit a maximum of 5 path segments (A, B, C) when we consider margins. The minimum height of a 6 letter sequence has 3 As and 3 Bs (or Cs). This gives us $9 > 7$.

Next, we will calculate the number of unique paths in each combination of letters when $n = 7$ by multiplying the matrix component with the path component:

For ***B***: $(1)(1) = 1$
For ***AB***: $(2)(364) = 728$
For ***AAB***: $(1)(547) = 547$
For ***ABB***: $(1)(58) = 58$
For ***AABB***: $(2)(42) = 84$
For ***AAABB***: $(1)(11) = 11$
For ***AABBB***: $(1)(1) = 1$
For ***AAC***: $(1)(547) = 547$
For ***ABC***: $(2)(58) = 116$
For ***AABC***: $(4)(42) = 168$
For ***AAABC***: $(2)(11) = 22$
For ***BBC***: $(1)(6) = 6$
For ***ABBC***: $(4)(6) = 24$
For ***AABBC***: $(8)(1) = 8$
For ***AAACC***: $(1)(11) = 11$
For ***AABCC***: $(3)(1) = 3$
Lastly: $3^n = 3^7 = 2187$
$1 + 728 + 547 + 58 + 84 + 11 + 1 + 547 + 116 + 168 + 22 + 6 + 24 + 8 + 11 + 3 + 2187 = 4522$
Which is the correct term for $n = 7$.

H. Acknowledgements

We would like to thank Mr Glenn Tay, our teacher mentor, for giving us this opportunity to explore and learn a whole new world of mathematics, with lessons far and few between in our regular syllabus. We thank him for his constant guidance and oversight, spurring us on and ensuring that timelines were met.

We would also like to thank Dr. Chua, our professor, who would, no matter the time, review our work. We are grateful for his guidance in how we could translate our mathematical ideas and solutions into words on paper in an understandable and intuitive manner. Lastly, we thank him for his mathematical insight, providing us with unique points of view and differing ways to attack our problems, and ensuring the coherence of our paper.

I. References

A006192: Number of nonintersecting (or self-avoiding) rook paths joining opposite corners of 3 X n board. (n.d.). OEIS. Retrieved 2021, from https://oeis.org/A006192

Abbott, H.L., & Hanson, D. (1978). *A Lattice Path Problem*. Retrieved 2021, from https://www.cs.purdue.edu/homes/spa/papers/abbott78.pdf

J. Appendix

```
import math
n = 2
m = 2
## n is horizontal length and m is vertical height ##
pathnumber = []
digitnumber = n + m
while digitnumber < (n+1)*(m+1):
 setnumber = 0
 while setnumber < (4**digitnumber):
   pastlocation = [[0,0]]
   i = 0
   ign = 0
   while i < digitnumber:
     recentlocation = pastlocation[-1]
     currentmove = (math.floor(setnumber/(4**i)))%4
     if currentmove == 0:
       newlocation = [recentlocation[0]+1,recentlocation[1]]
     if currentmove == 1:
       newlocation = [recentlocation[0],recentlocation[1]+1]
     if currentmove == 2:
       newlocation = [recentlocation[0]-1,recentlocation[1]]
     if currentmove == 3:
       newlocation = [recentlocation[0],recentlocation[1]-1]
     pastlocation.append(newlocation)
     if newlocation[0] > n or newlocation[0] < 0 or newlocation[1] > m or
newlocation[1] < 0:
       ign = 1
       break
     if recentlocation == [n,m]:
       ign = 1
       break
     if pastlocation.count(newlocation) != 1:
       ign = 1
       break
     i += 1
   if pastlocation[-1] == [n,m] and ign == 0:
     pathnumber.append(str(setnumber) + " " + str(digitnumber))
   setnumber += 1
```

```
 digitnumber += 2
print(len(pathnumber))
print(pathnumber)
```