Section 4b: Combinatorics – placement and more counting

Mathematics Admissions Test Questions

Question G from the 2010 Paper

G. The function f, defined for whole positive numbers, satisfies $f(1) = 1$ and also the rules

$$f(2n) = 2f(n),$$
$$f(2n+1) = 4f(n),$$

for all values of n. How many numbers n satisfy $f(n) = 16$?

(a) 3, (b) 4, (c) 5, (d) 6.

Solution:

G. The function f satisfies $f(1) = 1$ and also the rules

$$f(2n) = 2f(n), \qquad f(2n+1) = 4f(n).$$

The value 16 can be achieved by applying the first rule 4 times *or* by applying the first rule twice and the second rule once *or* by applying the second rule twice. However – for the second possibility – it matters what order the rules are applied. So we see the possibilities are:

$$\begin{aligned}
f(16) &= 2f(8) = 4f(4) = 8f(2) = 16f(1) = 16, && \text{[first rule four times]}\\
f(9) &= 4f(4) = 8f(2) = 16f(1) = 16, && \text{[second rule, first rule, first rule]}\\
f(10) &= 2f(5) = 8f(2) = 16f(1) = 16, && \text{[second rule, first rule, first rule]}\\
f(12) &= 2f(6) = 4f(3) = 16f(1) = 16, && \text{[second rule, first rule, first rule]}\\
f(7) &= 4f(3) = 16f(1) = 16, && \text{[second rule twice]}
\end{aligned}$$

There are 5 possible solutions ait matters and the **answer is (c).**

Question 2 from the 2012 paper

2. For **ALL APPLICANTS.**

Let

$$f(x) = x + 1 \qquad \text{and} \qquad g(x) = 2x.$$

We will, for example, write fg to denote the function "perform g then perform f" so that

$$fg(x) = f(g(x)) = 2x + 1.$$

If $i \geqslant 0$ is an integer we will, for example, write f^i to denote the function which performs f i times, so that

$$f^i(x) = \underbrace{fff\cdots f}_{i\text{ times}}(x) = x + i.$$

(i) Show that

$$f^2g(x) = gf(x).$$

(ii) Note that

$$gf^2g(x) = 4x + 4.$$

Find all the other ways of combining f and g that result in the function $4x + 4$.

(iii) Let $i, j, k \geqslant 0$ be integers. Determine the function

$$f^igf^jgf^k(x).$$

(iv) Let $m \geqslant 0$ be an integer. How many different ways of combining the functions f and g are there that result in the function $4x + 4m$?

Solution:

2. (i) [2 marks] Note that

$$f^2g(x) = f(f(g(x))) = f(f(2x)) = f(2x+1) = 2x+2;$$
$$gf(x) = g(x+1) = 2(x+1) = 2x+2.$$

(ii) [3 marks] Using the identity $f^2g = gf$ we see that

$$gf^2g = g(f^2g) = g(gf) = g^2f$$

and also that

$$gf^2g = gf(fg) = f^2g(fg) = f^2gfg$$

and finally that

$$f^2gfg = f^2(gf)g = f^2\left(f^2g\right)g = f^4g^2.$$

These four, f^4g^2, f^2gfg, gf^2g, g^2f are the only sequences that lead to $4x + 4$.

(iii) [4 marks] Note that

$$f^k(x) = x + k;$$
$$gf^k(x) = 2(x+k) = 2x + 2k;$$
$$f^jgf^k(x) = (2x+2k) + j = 2x + 2k + j;$$
$$gf^jgf^k(x) = 2(2x+2k+j) = 4x + 4k + 2j;$$
$$f^igf^jgf^k(x) = (4x+4k+2j) + i = 4x + 4k + 2j + i.$$

(iv) [6 marks] We need to consider the ways we can have $4k + 2j + i = 4m$ where $i, j, k \geqslant 0$. Clearly k can take the values $0 \leqslant k \leqslant m$ and then we need

$$2j + i = 4(m - k).$$

Then j can take values $0 \leqslant j \leqslant 2(m - k)$ – that is $2m - 2k + 1$ choices for j (given k). The choice of j then determines i. So the number of possible combinations is

$$\begin{aligned}\sum_{k=0}^{m}(2m - 2k + 1) &= (2m+1)\left(\sum_{k=0}^{m} 1\right) - 2\left(\sum_{k=0}^{m} k\right) \\ &= (2m+1)(m+1) - 2 \times \frac{1}{2}m(m+1) \\ &= (m+1)(m+1) \\ &= (m+1)^2.\end{aligned}$$

Question 5 from the 2007 paper

5. For **ALL APPLICANTS.**

Let $f(n)$ be a function defined, for any integer $n \geqslant 0$, as follows:

$$f(n) = \begin{cases} 1 & \text{if } n = 0, \\ (f(n/2))^2 & \text{if } n > 0 \text{ and } n \text{ is even}, \\ 2f(n-1) & \text{if } n > 0 \text{ and } n \text{ is odd}. \end{cases}$$

(i) What is the value of $f(5)$?

The *recursion depth* of $f(n)$ is defined to be the number of other integers m such that the value of $f(m)$ is calculated whilst computing the value of $f(n)$. For example, the recursion depth of $f(4)$ is 3, because the values of $f(2)$, $f(1)$, and $f(0)$ need to be calculated on the way to computing the value of $f(4)$.

(ii) What is the recursion depth of $f(5)$?

Now let $g(n)$ be a function, defined for all integers $n \geqslant 0$, as follows:

$$g(n) = \begin{cases} 0 & \text{if } n = 0, \\ 1 + g(n/2) & \text{if } n > 0 \text{ and } n \text{ is even}, \\ 1 + g(n-1) & \text{if } n > 0 \text{ and } n \text{ is odd}. \end{cases}$$

(iii) What is $g(5)$?

(iv) What is $g(2^k)$, where $k \geqslant 0$ is an integer? Briefly explain your answer.

(v) What is $g(2^l + 2^k)$ where $l > k \geqslant 0$ are integers? Briefly explain your answer.

(vi) Explain briefly why the value of $g(n)$ is equal to the recursion depth of $f(n)$.

Solution:

QUESTION 5: (i) [2 marks]

$$f(5) = 2f(4) = 2(f(2))^2 = 2\left(\left(f(1)^2\right)\right)^2 = 2\left(\left(2^2\right)^2\right) = 32.$$

(ii) [2 marks] As we had to calculate $f(4), f(2), f(1), f(0)$ on the way then $f(5)$ has recursion depth 4.

(iii) [2 marks]

$$g(5) = 1 + g(4) = 1 + 1 + g(2) = 1 + 1 + 1 + g(1) = 1 + 1 + 1 + 1 + g(0) = 4.$$

(iv) [3 marks] For any natural number k

$$g\left(2^k\right) = 1 + g\left(2^{k-1}\right) = \cdots = k + g\left(2^0\right) = k + g(1) = k + 1.$$

(v) [4 marks] For natural numbers $l > k \geqslant 0$

$$g\left(2^l + 2^k\right) = k + g\left(2^{l-k} + 1\right) = k + 1 + g\left(2^{l-k}\right) = k + 1 + l - k + 1 = l + 2.$$

(vi) [2 marks] In the definition of $g(n)$ a further 1 is added to previously calculated values at each stage whether n is even or odd; as $g(0) = 0$ then $g(n)$ is a measure of the number of previously calculated values, i.e. $g(n)$ equals the recursion depth.

Question 5 from the Specimen 2 paper issued in 2009

5. For **ALL APPLICANTS.**

An $n \times n$ square array contains 0s and 1s. Such a square is given below with $n = 3$.

$$\begin{matrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{matrix}$$

Two types of operation C and R may be performed on such an array.

- The first operation C takes the first and second columns (on the left) and replaces them with a single column by comparing the two elements in each row as follows; if the two elements are the same the C replaces them with a 1, and if they differ C replaces them with a 0.
- The second operation R takes the first and second rows (from the top) and replaces them with a single row by comparing the two elements in each column as follows; if the two elements are the same the R replaces them with a 1, and if they differ R replaces them with a 0.

By way of example, the effects of performing R then C on the square above are given below.

$$\begin{matrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{matrix} \xrightarrow{R} \begin{matrix} 0 & 1 & 0 \\ 1 & 1 & 0 \end{matrix} \xrightarrow{C} \begin{matrix} 0 & 0 \\ 1 & 0 \end{matrix}$$

(a) If R then C are performed on a 2×2 array then only a single number (0 or 1) remains.

(i) Write down in the grids on the next page the eight 2×2 arrays which, when R then C are performed, produce a 1.

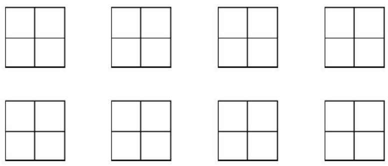
(ii) By grouping your answers accordingly, show that if $\begin{matrix} a & b \\ c & d \end{matrix}$ is amongst your answers to part (i) then so is $\begin{matrix} a & c \\ b & d \end{matrix}$.

Explain why this means that doing R then C on a 2×2 array produces the same answer as doing C first then R.

(b) Consider now a $n \times n$ square array containing 0s and 1s, and the effects of performing R then C or C then R on the square.

(i) Explain why the effect on the right $n-2$ columns is the same whether the order is R then C or C then R. [This then also applies to the bottom $n-2$ rows.]

(ii) Deduce that performing R then C on an $n \times n$ square produces the same result as performing C then R.



Solution:

5. (a) (i) C produces 1 from a 1×2 grid if both entries are equal. So a 2×2 grid produces a 1, after R then C, if the entries of each column agree (the top row below) or if the entries of each column disagree (the bottom row). So the eight 2×2 grids which produce a 1 (after R, then C) are

$$\begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} \qquad \begin{matrix} 0 & 1 \\ 0 & 1 \end{matrix} \qquad \begin{matrix} 1 & 0 \\ 1 & 0 \end{matrix} \qquad \begin{matrix} 1 & 1 \\ 1 & 1 \end{matrix}$$

$$\begin{matrix} 0 & 0 \\ 1 & 1 \end{matrix} \qquad \begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix} \qquad \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} \qquad \begin{matrix} 1 & 1 \\ 0 & 0 \end{matrix}$$

(a) (ii) The first, fourth, sixth, seventh grids are symmetric in the top-left-bottom-right diagonal and the second/fifth and third/eighth are reflections of one another about this diagonal. Alternatively, we can say that the grids listed above are those that contain an even number of 1s, in which case their reflections in the leading diagonal will also have an even number of 1s.

The effect of doing R then C on $\begin{matrix} \mathrm{a} & \mathrm{b} \\ \mathrm{c} & \mathrm{d} \end{matrix}$ is the same as doing C then R on $\begin{matrix} \mathrm{a} & \mathrm{c} \\ \mathrm{b} & \mathrm{d} \end{matrix}$.

This means that if the first effect is a 1 then so will is the second. Similarly if the first effect is a 0, and so it not amongst the above eight grids, then neither will its reflection be; hence doing C then R on the reflection and will also produce a 0.

(b) (i) If we consider the right $n-2$ columns of the $n \times n$ grid then C has no effect on them whatsoever, whether done first or second. The effect of R is to compare the top two rows, but this is the same effect whether done first or second.

(b) (ii) From the previous part the effect on the bottom $n-2$ rows, and right $n-2$ columns, is the same irrespective of order. From part (a) of the question the effects on the top-left 2×2 entries are also the same and so the order in which R and C are performed does not matter.

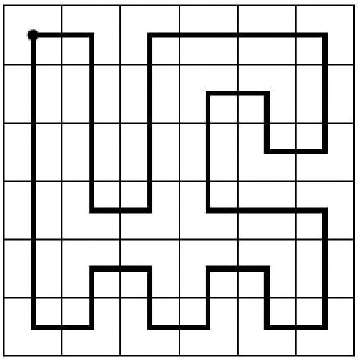
Question 5 from the 2009 Paper

5. For **ALL APPLICANTS.**

Given an $n \times n$ grid of squares, where $n > 1$, a *tour* is a path drawn within the grid such that:

- along its way the path moves, horizontally or vertically, from the centre of one square to the centre of an adjacent square;
- the path starts and finishes in the same square;
- the path visits the centre of every other square just once.

For example, below is a tour drawn in a 6×6 grid of squares which starts and finishes in the top-left square.



For parts (i)-(iv) it is assumed that n is *even*.

(i) With the aid of a diagram, show how a tour, which starts and finishes in the top-left square, can be drawn in any $n \times n$ grid.

(ii) Is a tour still possible if the start/finish point is changed to the centre of a different square? Justify your answer.

Suppose now that a robot is programmed to move along a tour of an $n \times n$ grid. The robot understands two commands:

- command R which turns the robot clockwise through a right angle;
- command F which moves the robot forward to the centre of the next square.

The robot has a program, a list of commands, which it performs in the given order to complete a tour; say that, in total, command R appears r times in the program and command F appears f times.

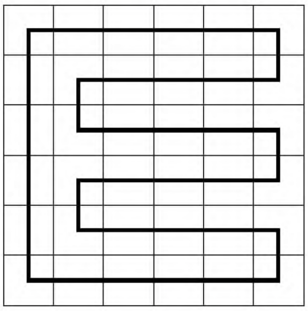
(iii) Initially the robot is in the top-left square pointing to the right. Assuming the first command is an F, what is the value of f? Explain also why $r + 1$ is a multiple of 4.

(iv) Must the results of part (iii) still hold if the robot starts and finishes at the centre of a different square? Explain your reasoning.

(v) Show that a tour of an $n \times n$ grid is not possible when n is odd.

Solution:

5. (i) [3 marks] A tour of an even by even grid can be produced by generalizing the sketch below.



By moving right along the first row, dropping a row, left to column two, dropping a row, and so, and then finally coming back all the way along the bottom row and coming up the leftmost column, we complete a tour. As n is even then this general E-shapes has $n/2$ "arms" and so it is the case that having descended all the rows the tour returns left along the bottom row.

(ii) [2 marks] Given it is possible to draw a tour which goes through all the squares, our new starting point is somewhere on the original tour and so a new tour can begin from there and proceed along the original tour to the top-left corner and then catch up on the part of the original tour that it had missed.

(iii) [3 marks] We have $f = n^2$ as the robot has to move through each of the squares of the grid without repetition and moves into a new square with each F command. As it starts in a corner going right it must return to that corner by moving up and so to go from travelling right to up means that robot has turned 270 degrees over all, or one right angle short of a number of whole turns.Equivalently $r + 1$ clockwise quarter turns have led to a whole number of turns in all and so $r + 1$ must be a multiple of 4 (there being four quarter turns in a whole turn).

(iv) [3 marks] As the robot still travels through all the squares we still have $f = n^2$. But if, for example, the robot set off going right along one of the flat parts of the E-like tour above then it would overall turn through 360 degrees and, instead, r would be a multiple of 4

(v) [4 marks] Suppose now that n is odd. Then $f = n^2$ also is odd. But as any tour begins and ends in the same place, for every move to the right there must be one to the left, and for every up there must be a down i.e. f must be even. Hence no tour is possible for odd n.

Year 12 Problem-Solving Course

Question 5 from the 2010 Paper

5. For **ALL APPLICANTS.**

This question concerns calendar dates of the form

$$d_1d_2/m_1m_2/y_1y_2y_3y_4$$

in the order day/month/year.

The question specifically concerns those dates which contain no repetitions of a digit. For example, the date 23/05/1967 is one such date but 07/12/1974 is not such a date as both $1 = m_1 = y_1$ and $7 = d_2 = y_3$ are repeated digits.

We will use the Gregorian Calendar throughout (this is the calendar system that is standard throughout most of the world; see below.)

(i) Show that there is no date with no repetition of digits in the years from 2000 to 2099.

(ii) What was the last date before today with no repetition of digits? Explain your answer.

(iii) When will the next such date be? Explain your answer.

(iv) How many such dates were there in years from 1900 to 1999? Explain your answer.

[The Gregorian Calendar uses 12 months, which have, respectively, 31, 28 or 29, 31, 30, 31, 30, 31, 31, 30, 31, 30 and 31 days. The second month (February) has 28 days in years that are not divisible by 4, or that are divisible by 100 but not 400 (such as 1900); it has 29 days in the other years (leap years).]

Solution:

5. (a) [2 marks] Let's consider dates of the form $d_1d_2/m_1m_2/20y_3y_4$. Clearly $m_1 = 1$ (to avoid repetition of 0). But then $m_2 = 0, 1$ or 2 each of which would be a repetition. Hence there are no such dates.

(b) [3 marks] As there are no such dates this century, let's consider dates of the form $d_1d_2/m_1m_2/19y_3y_4$. The last possible year is 1987; the last possible month is 06; and the last possible day is 25. This gives the date 25/06/1987.

(c) [6 marks] As there are no such dates this century, let's consider dates of the form $d_1d_2/m_1m_2/21y_3y_4$. Now $m_1 = 0$ (to avoid repetitions). Then $d_1 = 3$ (to avoid repetitions). But that leaves no possible value for d_1. Clearly there is no such date of the form $d_1d_2/m_1m_2/22y_3y_4$. For dates of the form $d_1d_2/m_1m_2/23y_3y_4$: if $m_1 = 1$ then $m_2 = 0$ and there is no possibility left for d_1. So $m_1 = 0$ and $d_1 = 1$. Of the dates $1d_2/0m_2/23y_3y_4$ the earliest possible year is 2345, the earliest possible month is 06, and the earliest possible day is 17. This gives the date 17/06/2345.

(d) [4 marks] Let's consider dates of the form $d_1d_2/m_1m_2/19y_3y_4$. Clearly $m_1 = 0$. If $d_1 = 3$, then $d_2 = 0$ or 1, either of which would be a repetition. Hence $d_1 = 2$. We therefore have dates of the form $2d_2/0m_2/19y_3y_4$. The remaining spaces can be filled with arbitrary distinct values from 3, 4, 5, 6, 7, 8, giving $6 \times 5 \times 4 \times 3 = 360$ possibilities; each such possibility is a valid date.

Question 5 from the 2011 Paper

5. For **ALL APPLICANTS.**

An $n \times n$ *grid* consists of squares arranged in n rows and n columns; for example, a chessboard is an 8×8 grid. Let us call a *semi-grid* of size n the lower left part of an $n \times n$ grid – that is, the squares located on or below the grid's diagonal. For example, Figure C shows an example of a semi-grid of size 4.

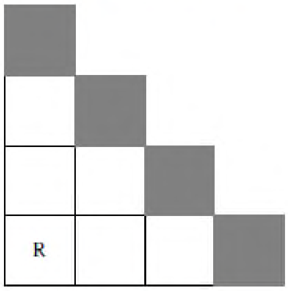


Figure C

Let us suppose that a robot is located in the lower-left corner of the grid. The robot can move only up or right, and its goal is to reach one of the *goal squares*, which are all located on the semi-grid's diagonal. In the example shown in Figure C, the robot is initially located in the square denoted with R, and the goal squares are shown in grey. Let us call a *solution* a sequence of the robot's moves that leads the robot from the initial location to some goal square.

(i) Write down all 8 solutions for a robot on a semi-grid of size 4.

(ii) Devise a concise way of representing the possible journeys of the robot in a semi-grid of size n. In your notation, which of the journeys are solutions?

(iii) Write down a formula for the number of possible solutions in a semi-grid of size n. Explain why your formula is correct.

Now let us change the problem slightly and redefine a goal square as any square that can be described as follows:

- the lower-left square is not a goal square;
- each square that is located immediately above or immediately to the right of a non-goal square is a goal square; and
- each square that is located immediately above or immediately to the right of a goal square is a non-goal square.

Furthermore, let us assume that, upon reaching a goal square, the robot may decide to stop or to continue moving (provided that there are more allowed moves).

(iv) With these modifications in place, write down all the solutions in a semi-grid of size 4, and all the solutions in a semi-grid of size 5.

(v) How many solutions are there now in a semi-grid of size n, where n is a positive integer? You may wish to consider separately the cases where n is even or odd.

Solution:

5. (i) [2 marks] All solutions in a semi-grid of size 4 are as follows:

$$RRR, \quad RRU, \quad RUR, \quad RUU, \quad URR, \quad URU, \quad UUR, \quad UUU.$$

(ii) [2 marks] Write R for a right-move and U for an up-move. A solution in a semi-grid of size n is then a sequence of $n-1$ letters, U or R.

(iii) [2 marks] Each solution consists of $n-1$ letters, and each letter can be chosen independently; thus there are 2^{n-1} solutions in total.

(iv) [3 marks] In a semi-grid of size 4 there are now ten solutions: R, U and the eight solutions from part (ii).

In a semi-grid of size 5: the same ten solutions.

(v) [6 marks] The modified definitions ensure that the goal squares are actually the squares on the diagonals of semi-grids of sizes $2, 4, 6, \ldots$ and so on, up to $n-1$ (if n is odd) or n (if n is even). Then, by using the result from part (iii) of this question, the total number of paths from the original location to a goal square can be obtained as follows. If $n = 2k$ then the sum is

$$2^{2-1} + 2^{4-1} + \cdots + 2^{2k-1} = \frac{1}{2}\sum_{i=1}^{k} 2^{2i}.$$

And in fact if $n = 2k+1$ then we arrive at the same sum.

As this is the sum of a geometric progression, we obtain

$$\frac{1}{2}\left(\frac{4\left(4^k-1\right)}{4-1}\right) = \frac{2}{3}\left(4^k-1\right).$$

This can also be written as $\frac{2}{3}\left(2^{2k}-1\right)$ or $\frac{2}{3}\left(2^{n-1}-1\right)$ when n is odd and $\frac{2}{3}\left(2^{n}-1\right)$ when n is even.

Question 5 from the 2012 Paper

5. For **ALL APPLICANTS.**

A particular robot has three commands:

F: Move forward a unit distance;
L: Turn left 90°;
R: Turn right 90°.

A *program* is a sequence of commands. We consider particular programs P_n (for $n \geqslant 0$) in this question. The basic program P_0 just instructs the robot to move forward:

$$P_0 = \mathbf{F}.$$

The program P_{n+1} (for $n \geqslant 0$) involves performing P_n, turning left, performing P_n again, then turning right:

$$P_{n+1} = P_n \, \mathbf{L} \, P_n \, \mathbf{R}.$$

So, for example, $P_1 = \mathbf{F\,L\,F\,R}$.

(i) Write down the program P_2.

(ii) How far does the robot travel during the program P_n? In other words, how many **F** commands does it perform?

(iii) Let l_n be the total number of commands in P_n; so, for example, $l_0 = 1$ and $l_1 = 4$.

Write down an equation relating l_{n+1} to l_n. Hence write down a formula for l_n in terms of n. No proof is required. **Hint:** consider $l_n + 2$.

(iv) The robot starts at the origin, facing along the positive x-axis. What direction is the robot facing after performing the program P_n?

(v) The left-hand diagram on the opposite page shows the path the robot takes when it performs the program P_1. On the right-hand diagram opposite, draw the path it takes when it performs the program P_4.

(vi) Let (x_n, y_n) be the position of the robot after performing the program P_n, so $(x_0, y_0) = (1, 0)$ and $(x_1, y_1) = (1, 1)$. Give an equation relating (x_{n+1}, y_{n+1}) to (x_n, y_n).

What is (x_8, y_8)? What is (x_{8k}, y_{8k})?

Solution:

5. (i) [1 mark] From the recursive definition

$$\begin{aligned} P_0 &= \mathbf{F}, \\ P_1 &= P_0\,\mathbf{L}\,P_0\,\mathbf{R} = \mathbf{F\,L\,F\,R} \\ P_2 &= P_1\,\mathbf{L}\,P_1\,\mathbf{R} = \mathbf{F\,L\,F\,R\,L\,F\,L\,F\,R\,R}. \end{aligned}$$

(ii) [1 mark] Say there are f_n commands $\mathbf{F}$ in P_n. As $P_{n+1} = P_n\,\mathbf{L}\,P_n\,\mathbf{R}$ then $f_{n+1} = 2f_n$. As $f_0 = 1$ then $f_n = 2^n$.

(iii) [3 marks] Let l_n denote the total number of commands in P_n. As $P_{n+1} = P_n\,\mathbf{L}\,P_n\,\mathbf{R}$ then

$$l_{n+1} = l_n + 1 + l_n + 1 = 2l_n + 2.$$

If we set $m_n = l_n + 2$ (following the hint) then we see

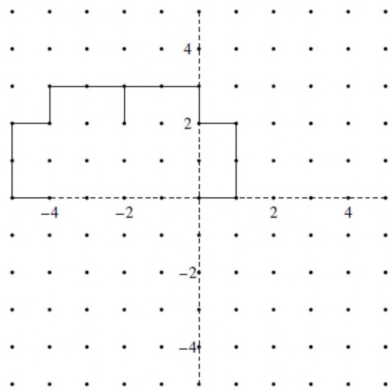
$$m_{n+1} = l_{n+1} + 2 = 2l_n + 4 = 2\,(l_n + 2) = 2m_n.$$

As $m_0 = l_0 + 2 = 3$ then $m_n = 3 \times 2^n$ and

$$l_n = m_n - 2 = 3 \times 2^n - 2.$$

(iv) [1 mark] The robot again faces along the positive x-axis after each P_n because each P_n contains as many $\mathbf{L}$s as it does $\mathbf{R}$s.

(v) [3 marks] P_4 is sketched below:



(vi) [6 marks] After performing P_n the robot sits at (x_n, y_n) facing "east". Turning left it now faces "north" so what otherwise would have led to a movement of (x_n, y_n) instead achieves $(-y_n, x_n)$. So from the recursion $P_{n+1} = P_n\,\mathbf{L}\,P_n\,\mathbf{R}$ we can see that

$$(x_{n+1}, y_{n+1}) = (x_n, y_n) + (-y_n, x_n) = (x_n - y_n, x_n + y_n)\,.$$

Note then that

$$\begin{aligned} (x_{n+2}, y_{n+2}) &= (x_{n+1} - y_{n+1}, x_{n+1} + y_{n+1}) = (-2y_n, 2x_n)\,; \\ (x_{n+4}, y_{n+4}) &= (-2y_{n+2}, 2x_{n+2}) = (-4x_n, -4y_n)\,; \\ (x_{n+8}, y_{n+8}) &= (-4x_{n+4}, -4y_{n+4}) = (16x_n, 16y_n)\,. \end{aligned}$$

Hence

$$(x_8, y_8) = (16, 0) \qquad \text{and} \qquad (x_{8k}, y_{8k}) = \left(16^k, 0\right).$$

Question 5 from Specimen Paper I issued in 2009

Songs of the Martian classical period had just two notes (let us call them x and y) and were constructed according to rigorous rules:

I. the sequence consisting of no notes was deemed to be a song (perhaps the most pleasant);

II. a sequence starting with x, followed by two repetitions of an existing song and ending with y was also a song;

III. the sequence of notes obtained by interchanging xs and ys in a song was also a song.

All songs were constructed using those rules.

(i) Write down four songs of length six (that is, songs with exactly six notes).

(ii) Show that if there are k songs of length m then there are $2k$ songs of length $2m+2$. Deduce that for each natural number there are 2^n songs of length $2^{n+1}-2$.

Songs of the Martian later period were constructed using also the rule:

IV. if a song ended in y then the sequence of notes obtained by omitting that y was also a song.

(iii) What lengths do songs of the later period have? That is, for which natural numbers n is there a song with exactly n notes? Justify your answer.

Solution:

5. (i) The only way to create longer songs, from previous ones, is by Rule II. xy is a song of length two (as it can be made from Rule II as x-no notes-no notes-y) and by Rule III yx is also a song. With these two songs and using Rule II again, we can produce two songs

$$xxyxyy \text{ and } xyxyxy$$

of length six, and applying Rule III we also see that

$$yyxyxx \text{ and } yxyxyx$$

are songs.

(ii) As was noted in (i) longer songs may only be produced from shorter ones using Rule II. Given any song s of length m then the song $xssy$ is also a song. By this means, k songs of length m create k new songs of length $2m+2$. The only way to produce further songs of length $2m+2$ is to swap the xs and ys in the newly made songs by Rule III. So $y\tilde{s}\tilde{s}x$ will also be a song of length $2m+2$ where $\tilde{s}$ denotes the song s with all xs and ys swapped. As the new batch of songs end in an x and the previous batch ended in a y then the second batch of length $2m+2$ songs contains none of the first batch. In total, then, we have $2k$ new songs of length $2m+2$.

Each time this process of generating new songs produces twice as many songs of the next allowable length as for the previous allowed length. As there is one song of the "noughth" allowed length there will be 2^n of the nth allowed length for $n \geqslant 0$. What is the nth allowed length? These allowed lengths follow the rule

$$\begin{aligned}
\text{length}_0 &= 0 \\
\text{length}_1 &= 2 \times 0 + 2 = 2 \\
\text{length}_2 &= 2 \times 2 + 2 = 2^2 + 2 \\
\text{length}_3 &= 2 \times \left(2^2 + 2\right) + 2 = 2^3 + 2^2 + 2 \\
&\vdots \\
\text{length}_n &= 2^n + 2^{n-1} + \cdots + 2.
\end{aligned}$$

This length_n is a geometric sum with n terms, common ratio 2, and first term 2, and so we have

$$\text{length}_n = \frac{2\left(2^n - 1\right)}{(2-1)} = 2^{n+1} - 2.$$

[For those with knowledge of *mathematical induction* then part (ii) could be attempted using this, for full marks, but no presumption about such knowledge has been made.]

(iii) For any positive whole number n, it is possible to produce a song of greater length using Rules I, II, III, because we can find a k such that $2^{k+1} - 2 > n$. Set $N = 2^{k+1} - 2$. If this longer (length N) song ends in a y we may reduce its length using Rule IV to make another song one note shorter. On the other hand, if the song of length N ends in an x we can swap all the xs and ys by Rule III to produce a song, also of length N, which now ends in an y. Removing the final y by Rule IV we again have produced a song of length $N-1$.

Repeating this process, one note at a time, we eventually produce a song of length n, and so songs of all possible lengths exist in the Martian later period.