Oxford Mathematics Team Challenge
Lock-in Round Question Booklet

Saturday, 8$^{\text{th}}$ March 2025

Instructions

1. Do not open the paper until the invigilator tells you to do so.
2. **Format.** This round contains 4 questions. For your answers, write them on sheets of paper (lined or blank, either is fine) and mark the question-part you are answering on the margin. You must also clearly write your Team ID at the top of the page. You do *not* need to write your team name.

   The questions are long-answer, so you may be required to give detailed explanations, brief descriptions, or mathematical working. The questions may indicate the level of depth you should offer, but you should always exercise your judgement in giving an appropriate level of depth to your answer.
3. **Time limit.** 45 minutes. You may not write anything on any paper you will submit after the allotted time has expired.
4. **No calculators, squared paper or measuring instruments.** Lined paper and blank paper for rough working is allowed. You may use a pen or pencil to preference. Other mediums for working (e.g., digital devices, whiteboards, thingamabobs) are strictly forbidden.

   The points for each question are in the bottom right of the cells in the Answer Sheet, as well as at the end of the questions in the Question Booklet, and are marked in [square brackets].
5. Don't expect to complete the whole paper in the time! The later parts are worth more marks but are generally harder, and they may build up on previous parts of the question.
6. You are also encouraged to think deeply, rather than to guess.
7. To accommodate the online version of the competition, please do not discuss or distribute the paper online until 08:00 BST on Monday 24$^{\text{th}}$ March.
8. Good luck, and enjoy! ☺

1 Circle packing

In this question, we will look at the circle-packing problem. In the circle-packing problem, we are given a shape and have to “pack” a certain number of congruent circles inside it: we must place the circles without overlap in the interior of the shape. The goal of the problem is to maximise the radius of the circles.

(a) Figure 1 shows a packing of six circles in a larger circle of radius 3.

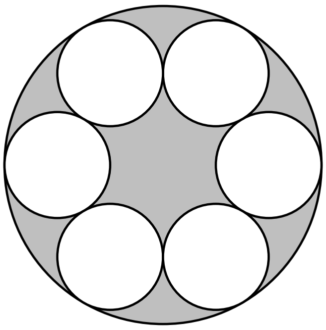

Figure 1

Carefully find the radius of the smaller circles. [3]

The packing of the six circles in Figure 1 is *optimal*, meaning that there is no way to fit six congruent circles of a larger radius into a circle of radius 3.

(b) Construct an optimal packing of seven circles in a circle of radius 3. [1]

Given two packings of n circles inside a shape, we say that these packings are *equivalent* if we can essentially transform one packing into the other through a series of transformations which slide the circles or rotate the whole packing.

(c) Are all optimal packings of six circles in a circle equivalent? Explain your answer. [3]

(d) An inconspicuous aside: Figure 2 depicts a triangle with side lengths 13, 13 and 10.

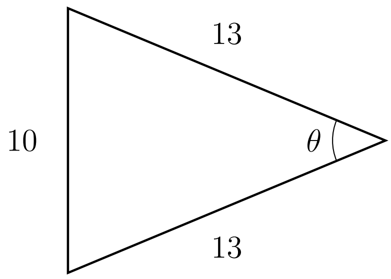

Figure 2

Show that θ, the angle between the sides of length 13, is approximately $45°$. Justify your working clearly. [3]

(e) Figure 3 shows an optimal packing of nine circles in a circle of radius 3.

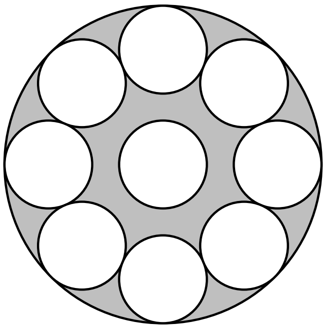

Figure 3

Find the radius r of the smaller circles in terms of $\sin(22.5°)$. [5]

(f) Now suppose the central circle in Figure 3 was enlarged so it is just touching the surrounding circles, which is depicted in Figure 4.

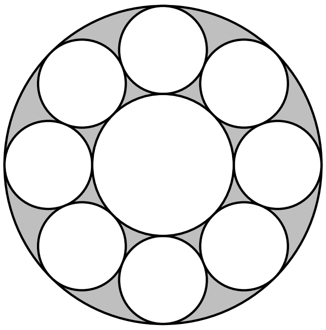

Figure 4

Let the central circle have radius R. Estimate the values of r and R. *[Hint: use (d) and (e).]* [5]

(g) Hence, or otherwise, verify that the ratio of the areas between the central circle and the smaller circles equals 2.6 when rounded to the nearest tenth. [2]

(h) Andrew wants to make a wedding cake by following his cake-making procedure:

Step A: Bake a cylindrical base with both radius and height 3 feet.

Step B: On the top of the cake, draw circular outlines as in Figure 4.

Step C: For each circular outline produced from the last step B, repeat this process, setting the radius and height in step A to the outline's radius.

In terms of R, what will the height of Andrew's cake be? [3]

(i) Find an expression for the volume of Andrew's cake in terms of R and r. [5]

Question 1 is out of 30 points

2 Integer partitions

In this question, we will explore *partition theory.* A *partition* of a positive integer n is a way of writing n as the sum of positive integers (called the *parts*), irrespective of the order of the sum. For example, $5+2+1$ is a partition of 8; this is the same partition as $2+1+5$, however a different partition of 8 is e.g. $3+3+2$. A sum with only one part counts as a partition, e.g. 5 is a partition of 5.

(a) Write down all the partitions of 4. [1]

(b) (i) A partition is called *distinct* if no part is repeated in the sum. For example, $5+2+1$ is a distinct partition of 8, whereas $3+3+2$ is not.

Write down all distinct partitions of 7. [2]

(ii) A partition is called *odd* if it only contains odd parts. For example, $5+3$ is an odd partition of 8, whereas $5+2+1$ is not.

Write down all odd partitions of 7. [2]

(c) For a positive integer n, let $p(n)$ be the number of partitions of n. By convention we also say $p(0)=1$.

(i) Calculate $p(5)$. [1]

(ii) Explain why $p(n+1) > p(n)$ for all $n \geq 1$. [2]

(iii) We also write $p_d(n)$ as the number of distinct partitions of n, and $p_o(n)$ as the number of odd partitions of n. Similarly, $p_d(0) = p_o(0) = 1$.

Calculate $p_d(5)$ and $p_o(5)$. [2]

We can draw out a partition using a Ferrer diagram, which represents the partition as a collection of dots, descending in size of the rows. Figure 5 shows two examples:

$6+3+3+2+1$ $\qquad$ $5+4+3+1+1+1$

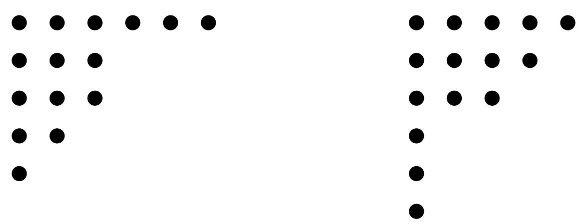

Figure 5

For any partition, we can create its *conjugate* by reflecting the Ferrer diagram over the diagonal. For example, the partitions in Figure 5 are conjugates of each other. We also say that a partition is *self-conjugate* if its conjugate is itself.

(d) (i) Give an example of a self-conjugate partition of 6. [1]

(ii) Give an example of a self-conjugate partition of 7. [1]

(iii) Explain why the number of partitions with six parts is the same as the number of partitions with the largest part equal to 6. [4]

(iv) Explain why the number of partitions of n into distinct odd parts is the same number of partitions of n into self-conjugate parts. [4]

(e) Let's return to $p_o(n)$ and $p_d(n)$.

(i) Consider the infinite products

$$A = (1+x)(1+x^2)(1+x^3)\cdots$$
$$B = (1+x+x^2+\cdots)(1+x^3+x^6+\cdots)(1+x^5+x^{10}+\cdots)\cdots$$

Explain and justify how the coefficients of x^n in A and B relate to $p_d(n)$ and $p_o(n)$. [5]

(ii) Show that $p_d(n) = p_o(n)$. *[Hint: consider* $\dfrac{1-x^{2k}}{1-x^k}$ *for each positive integer k.]* [5]

Question 2 is out of 30 points

3 Random tic-tac-toe

Two robots, Xeep and Obot, play a game of tic-tac-toe (a.k.a. noughts and crosses) on a 3×3 grid. Xeep, who goes first, marks its squares with Xs, and Obot marks with Os. They take turns choosing a random unfilled square to mark. Each robot is equally likely to choose any one of the unfilled squares on its turn. A robot wins by being the first to achieve a 3-in-a-row of their own symbols, either horizontally, vertically, or diagonally. If the grid gets filled completely with no such 3-in-a-row, then the game ends in a draw.

We give the robots additional instructions so that even if a robot has won before the grid is completely filled, the robots will continue to make random moves until the grid is completely filled. We shall call the final completely-filled grid of this game the *end grid*. Note that every possible end grid has 5 Xs and 4 Os.

The following questions lead to finding the probability of a draw.

(a) Briefly describe how you can determine whether or not a game of tic-tac-toe ended in a draw if you are provided the end grid of the game. [2]

(b) Given an end grid of a game that didn't end in a draw, is it always possible to determine who the winner was? Briefly explain. [3]

(c) There are exactly 126 possible end grids. What is the probability that a game of random tic-tac-toe ends in a draw? Express your answer as a simplified fraction. *[Hint: find the number of end grids that could result from a drawn game.]* [7]

The following questions lead to finding the probability that Xeep wins, or that Obot wins.

(d) How many end grids contain a 3-in-a-row of Os, but not a 3-in-a-row of Xs? [4]

(e) How many end grids contain both a 3-in-a-row of Os and a 3-in-a-row of Xs? [4]

(f) Consider a variant of tic-tac-toe called *ric-rac-roe*, where parts of the grid are shaded grey as shown in Figure 6. In this variant, Xeep can only make moves in the white squares, and Obot can only make moves in the grey squares. Both robots are still equally likely to choose any one of its legal moves for each turn. What is the probability that Xeep wins in ric-rac-roe? Express your answer as a simplified fraction. *[Hint: consider the probability that Obot can achieve a 3-in-a-row in three moves.]* [6]

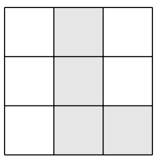

Figure 6

(g) What is the probability that Xeep wins a game of random tic-tac-toe? What is the probability of Obot winning? Express your answers as simplified fractions. [4]

Question 3 is out of 30 points

This page is intentionally left (mostly) blank

4 FRACTRAN

FRACTRAN is a Turing-complete programming language invented by John Conway. Turing-complete means it can perform a lot of complex tasks, and by the end you'll be able to write some programs!

A FRACTRAN program consists of a finite list of positive fractions. The program acts as a function that takes in a positive integer input, and it will output a positive integer according to the following procedure:

1. Given an integer n as the current input, the program looks for the first fraction in the list whose denominator is a factor of n.
2. If the program finds this fraction, which we write as $\frac{a}{b}$, then it repeats step 1 with $n \cdot \frac{a}{b}$ as the new current input.
3. If there is no such fraction in the list, then the program outputs its current input and stops running.

As an example, consider the FRACTRAN program $\left[\frac{5}{3}, \frac{5}{2}\right]$. Let's run this with 6 as input. The first fraction whose denominator divides 6 is $\frac{5}{3}$, so we repeat step 1 with $6 \cdot \frac{5}{3} = 10$ as the new input. The first fraction whose denominator divides 10 is $\frac{5}{2}$, so we repeat step 1 with $10 \cdot \frac{5}{2} = 25$ as the new input. No fraction in the list has a denominator dividing 25, so we output 25 and stop. When given 6 as input, our program outputs 25.

(a) We can record each step of a FRACTRAN program in a *log*, which is a list of integers starting with the input and ending with the output, with all intermediate steps in between in order. When we input 6 into the program $\left[\frac{5}{3}, \frac{5}{2}\right]$, our log will be $(6, 10, 25)$.

Write the logs of $\left[\frac{5}{3}, \frac{5}{2}\right]$ when ran on each of the following inputs: 4, 18, and 24. [3]

(b) When $\left[\frac{5}{3}, \frac{5}{2}\right]$ runs on input $2^x \cdot 3^y$, where x and y are non-negative integers, what is the output? Express your answer in terms of x and y. [1]

(c) According to the rules of FRACTRAN stated above, we allow our programs to contain unsimplified fractions. In fact, unsimplified fractions can lead to different behaviour from their simplified equivalents. To demonstrate this, consider the two programs $\left[\frac{2}{3}\right]$ and $\left[\frac{4}{6}\right]$. Find a positive integer input such that these programs produce different outputs when run on that input. [2]

(d) It is possible for a FRACTRAN program to run forever when given a certain input, never stopping and therefore never outputting anything. Write a FRACTRAN program and find a positive integer input such that this program will run forever when given this input. [2]

(e) Match each FRACTRAN program listed on the left to exactly one description listed on the right. The program should achieve the specified output when given the specified input, where x and y are general non-negative integers. Express your answer as pairs of one number and one letter for each match.

1. $\left[\frac{1}{2}\right]$
2. $\left[\frac{9}{2}\right]$
3. $\left[\frac{3}{6}, \frac{9}{2}\right]$
4. $\left[\frac{3}{10}, \frac{5}{2}, \frac{1}{5}\right]$
5. $\left[\frac{15}{10}, \frac{3}{5}, \frac{10}{2}\right]$

A. input 2^x, output 3^{x+1}
B. input $2^x \cdot 3^y$, output 3^{2x+y}
C. input 2^{x+1}, output 9
D. input $2^x \cdot 5^y$, output 5^y
E. input 2^x, output $3^{x/2}$ if x is even and $3^{(x-1)/2}$ if x is odd [5]

(f) Derrick claims that he created a FRACTRAN program such that for any positive integer input, it will output double the input. Briefly explain why Derrick's claim is impossible. [2]

(g) Patrick claims that he created a FRACTRAN program that outputs 1 when given any positive integer input. Briefly explain why Patrick's claim is impossible. [3]

(h) Write a FRACTRAN program such that for any positive integer x, an input of 2^x results in an output of 3 if x is even and an output of 5 if x is odd. [3]

(i) Write a FRACTRAN program such that for any non-negative integers x, y, and z, an input of $2^x \cdot 3^y \cdot 5^z$ results in an output of 7^m, where $m = \max\{x, y, z\}$, i.e. the maximum value of x, y, and z. [4]

(j) Write a FRACTRAN program such that for any non-negative integers x and y, an input of $2^x 3^y$ results in an output of 5^{xy}. *[Hint: consider a program whose log begins with* $(2^x \cdot 3^y, 2^{x-1} \cdot 3^y \cdot 11, 2^{x-1} \cdot 3^{y-1} \cdot 5 \cdot 7 \cdot 11, \dots, 2^{x-1} \cdot 5^y \cdot 7^y \cdot 11, 2^{x-1} \cdot 5^y \cdot 7^y, \dots)$.*]* [5]

Question 4 is out of 30 points