MATH36032 Coursework One

Student ID: 10098348

March 8, 2021

1 Question One

The aim of this question is to approximate the Euler-Mascheroni constant, γ, by a rational number, p/q where p and q are natural numbers. The Euler-Mascheroni constant is defined by

$$\gamma = \lim_{n\to\infty}\Big(-ln(n) + \sum_{k=1}^{n}\frac{1}{k}\Big).$$

Up to 15 decimal places, it has approximate value 0.577215664901533. Given an integer N, we want to find natural numbers p and q such that $|p/q - \gamma|$ is minimal amongst all positive integers p and q with $p + q \leq N$.

To do this we defined a function in Matlab called AppEM, which takes an integer N as input and returns two integers p, q such that p/q is the best approximation of γ satisfying certain conditons. Aside from the already stated conditions (p and q are natural numbers and $p+q \leq N$), we also want that if there is more than one pair which provide the best estimate then the function will return the pair $[p, q]$ with smallest sum.

The function works as follows, first we define emconst as the approximate value of γ to 15 decimal places and we give p and q initial value 1. We also define a variable absdiff which will later be used to store the value of $|emconst - p/q|$ but first we initialise it with a high value that isn't important.

Now we start two for loops with variables $p1$ and $q1$, both of which run from 1 to N, with the $q1$ loop inside the $p1$ loop. We calculate $p1/q1$, an approximation for the E-M constant, and store it in a variable app. Then we calculate the absolute difference between the approximation and emconst and store it as appdiff. If this value is less than absdiff and $p + q \leq N$, then we use an if statement and set the value of absdiff equal to appdiff and p and q equal to $p1$ and $q1$ respectively. If, instead, appdiff = absdiff and $p+q \leq N$

and $p1 + q1 < p + q$, we set $p = p1$ and $q = q1$. If appdiff > absdiff, then all values remain unchanged.

For input $N = 2021$, we get output

```
>> [p,q] = AppEM(2021)

p =

   228


q =

   395
```

2 Question Two

The goal of this question is to find the smallest MyLucky number greater than or equal to a given number. A MyLucky number, n, is defined by two conditions:

- all of n's prime factors are odd and distinct (and therefore at least two prime factors)
- if p is a prime factor of n, then $p - 1$ divides $n - 1$.

To solve this problem, we defined two functions. The first, isMyLuckynum, tested if a given number was MyLucky and the second, MyLuckynum, used the first and found the smallest MyLucky number greater than or equal to the given number.

The function isMyLuckynum takes a double as input and outputs a logical - true if the input is MyLucky and false if it is not. First, for input x, use the inbuilt factor function to create an array of factors of x, and store it in f. We then check if the factors are distinct by checking if the length of the array is equal to the length of the vector containing the unique elements of f and also if there is more than one factor. If either of these conditions is not met the function outputs false. If they are, we start a for loop and use it check if a factor is even or if for a factor p, $p - 1$ does not divide $x - 1$, using Matlab's inbuilt mod function and if they are then we set the output to false and break out of the loop. If all the factors are odd and divide $x - 1$, then we set the output to true.

The function MyLuckynum takes a double, N, as input and outputs a double, n, where n is the smallest MyLucky number greater than or equal to N. The function makes use of *isMyLuckynum* and works as follows. We use a while loop so that whilst N is not MyLucky (i.e. isMyLuckynum outputs false) then we increase N by 1. If N is MyLucky, or when it eventually equals a MyLucky number after passing through the while loop. we set $n = N$ and output.

With input $N = 2021$, we get output

```
>> n=MyLuckynum(2021)

n =

        2465
```

Through research online, I came across the concept of Carmichael numbers and Korselt's Criterion[1], which gives an equivalent definition for Carmichael numbers, and the conditions in Korselt's Criterion are precisely the same as those for MyLucky numbers. Looking at the OEIS sequence for Carmichael numbers we see that the smallest Carmichael (equivalently, MyLucky) number greater than or equal to 2021 is 2465[2], which is what we got as output from the function MyLuckynum.

3 Question Three

The problem here, is, given an integer N, to find an integer n such that n^2 is a beautiful square. A beautiful square number is a number whose square consists of all non-zero digits and each digit appears only once. To do this we wrote a function Beautisqnum. The function Beautisqnum makes use of a function dig2num which is the same as the code from labdemo5.1[3], just turned into a function. The steps behind the function Beautisqnum are explained below.