Binary Exercise Answers — Anakin Dey

Last Edited on 2/13/23 at 16:09

Line 2 of Algorithm-M

This is essentially adding a leading digit of 0. This digit $a[n+1]$ will act as a flag. While it is 0, we haven't finished enumerating through all numbers $a[n]\cdots a[1]$. For sake of example, suppose we are enumerating base 10, so $m[j]=10$ for all $1 \leq j \leq n$, and we just printed our last number $999\cdots 99$. We then enter the **while** loop to begin rolling over the digits. Since we have all 9's, we will always satisfy $a[j]=m[j]-1$. Thus the loop will end when $j=n+1$ since we will have $a[n+1]=0$ but $m[n+1]=2$ so $a[n+1]\neq m[j+1]-1$. So then we exit the **while** loop, $j=n+1$, and so we **return** and are done. If we did not have these auxiliary variables $a[n+1]$ and $m[n+1]$, we would not be able to so cleanly check when we are done enumerating numbers.

Prove that Γ_n generates all binary strings 0 to 2^n-1

⟨⟨ TODO: Proof by induction ⟩⟩

Line 2 of Algorithm-G

$a[0]$ acts as a parity bit. This bit is mainly there to clean the code up the check when to flip the last bit $a[1]$. This bit flips every time we make the loop (line 5). It turns out that whenever $a[0]=1$, we have that $a[1]=0$ and vice versa. So the **minimum** $j \geq 1$ such that $a[j-1]=1$ is $j=1$. So adding $a[0]$ and flipping the bit every iteration of the **while** loop allows us to cleanly check when to flip the bit $a[1]$.

1 Generating the modular sequence ([Knu11] Chapter 7.2.1.1 Exercise 77)

⟨⟨ TODO ⟩⟩

References

[Knu11] Donald E. Knuth. *The Art of Computer Programming, Volume 4A: Combinatorial Algorithms, Part 1*. 1st. Addison-Wesley Professional, 2011. ISBN: 0201038048.