Assignment - 2

RISHABH KUMAR
2021CS10103

Question 1:

For two binary numerals $a = a_m a_{m-1} a_{m-2} \cdots a_1 a_0$ and $b = b_n b_{n-1} \cdots b_1 b_0$ we define mul(a,b) gives $c_p c_{p-1} \cdots c_1 c_0$.

Where, a and b are representing natural numbers $\sum_{i=0}^{m} 2^i a_i$ and $\sum_{i=0}^{n} 2^i b_i$ respectively.

Specification

$$\sum_{i=0}^{p} 2^i c_i = \left(\sum_{i=0}^{m} 2^i a_i\right)\left(\sum_{i=0}^{n} 2^i b_i\right)$$

let us define functions 'lowest' and 'higher' which give the lowest bit and rest of the bits of a binary numeral respectively.

For any binary numeral, $a = a_m a_{m-1} \cdots a_1 a_0$ with $m > 0$.

lowest (a) $= a_0$ and higher(a) $= a_m a_{m-1} \cdots a_1$

Assume we have a function which put zero behind a binary numeral, like for $a = a_m a_{m-1} \cdots a_1 a_0$.

put_zero (a) $= a_m a_{m-1} \cdots a_1 a_0 0$

We have given a function add (a,b) which do addition of two any size binary numeral.

Then, we can define the '**multiplication function**' as follows: (recursively)

$$\text{mul}(a,b) = \begin{cases} 0, & \text{if } b \text{ has no digits} \\ \text{add}(a, p), & \text{if lowest}(b) \text{ is } 1 \\ p, & \text{if lowest}(b) \text{ is } 0. \end{cases}$$

where, p = mul(put-zero(a), higher(b))

Question-2

Proof of Correctness:

The helper functions 'lowest', 'higher' and 'put_zero' are correct essentially by definition.

And the helper function add(a,b) is given.

where, add(a,b) give binary numeral which represent the natural number $\left(\sum_{i=0}^{m} 2^i a_i + \sum_{i=0}^{n} 2^i b_i\right)$

For proving mul(a,b) function by strong induction -

The base case: when b has no digit or say have 0 value -

then we have mul(a,0) = 0

$$\left(\sum_{i=0}^{m} 2^i a_i\right)\cdot(0) = 0$$

Induction hypothesis: Assume that $\forall$ a (a binary numeral) and for all $n<k$, $b = b_n b_{n-1} \cdots b_1 b_0 \rightarrow (n+1)$ digits

mul(a,b) = $c_p c_{p-1} \cdots c_0$ and $\left(\sum_{i=0}^{m} 2^i a_i\right)\left(\sum_{i=0}^{n} 2^i b_i\right) = \left(\sum_{i=0}^{p} 2^i c_i\right)$.

Induction case: Let $n=k$, $k \geq 1$. $\rightarrow$ (k+1) digits

Subcase 1: lowest(b) = b_0 = 1

$$\text{mul}(a,b) = \text{add}(a, \text{mul}(a_m a_{m-1} \cdots a_1 a_0 0, \underbrace{b_k b_{k-1} \cdots b_1}_{k\text{-digits}}))$$

$$= \sum_{i=0}^{m} 2^i a_i + \left(\sum_{i=0}^{m} 2^{i+1} a_i\right)\cdot\left(\sum_{i=0}^{k} 2^i b_{i+1}\right)$$

$$= \sum_{i=0}^{m} 2^i a_i + 2\left(\sum_{i=0}^{m} 2^i a_i\right)\cdot\left(\sum_{i=0}^{k} 2^i b_{i+1}\right)$$

$$= \sum_{i=0}^{m} 2^i a_i \left(1 + \sum_{i=0}^{k} 2^{i+1} b_{i+1} \right)$$

$$= \sum_{i=0}^{m} 2^i a_i \left(2^0 b_0 + \sum_{i=1}^{k} 2^i b_i \right) \quad \text{as } b_0 = 1$$

$$\sum_{i=0}^{p} 2^i c_i = \left(\sum_{i=0}^{m} 2^i a_i \right) \left(\sum_{i=0}^{k=n} 2^i b_i \right)$$

<u>Subcase 2:</u>

$$\text{lowest}(b) = b_0 = 0$$

$$\text{mul}(a, b) = \text{mul}(a_m a_{m-1} \cdots a_1 a_0 0,\ b_k b_{k-1} \cdots b_1)$$

$$= \left(\sum_{i=0}^{m} 2^{i+1} a_i \right) \left(\sum_{i=0}^{k} 2^i b_{i+1} \right)$$

$$= \sum_{i=0}^{m} 2^i a_i \left(2^0 b_0 + \sum_{i=1}^{k} 2^i b_i \right) \quad \text{as } b_0 = 0$$

$$\sum_{i=0}^{p} 2^i c_i = \left(\sum_{i=0}^{m} 2^i a_i \right) \left(\sum_{i=0}^{k} 2^i b_i \right)$$

Question 4 :

After calculating Time and Space complexity of addition function given on moodle, we get:

TC of add (a,b) = $O(\max(m,n))$

SC of add (a,b) = $O((\max(m^2,n))^2)$

where, $a = a_m a_{m-1} \cdots a_1 a_0$ and

$b = b_n b_{n-1} \cdots b_1 b_0$

Now, for multiplication algorithm :-

$T(m,1) = 2(m+1) + O(\max(m+2, m+1))$

$= 2(m+1) + O(m+2)$ as 'b' has 2-digits & 'a' has m+1 digits.

Similarly, -

$T(m,2) = 3(m+1) + O(m+3) \times 2$

$\vdots$

$T(m,n) = (n+1)(m+1) + O(m+n+1) \times n$

$= mn + m+n + 1 + O(mn + n^2 + n)$

$T(m,n) = O(mn + n^2/2) \times 2$ # taking highest order.

$\therefore T(m,n) = O(mn + n^2)$

Now,

$$S(m,1) = 2(m+1) + O((\max(m+2, m+1))^2)$$

$\vdots$ no. of space = no. of bits stored.

$$S(m,2) = 3(m+1) + O((\max(m+3, m+2))^2)$$

$\vdots$

$$S(m,n) = (n+1)(m+1) + O((m+n)^2)$$

$$S(m,n) = O(m^2+n^2+mn)$$

Question 5:

(a) Define 'join_zero(a, i)' which put 'i' zeroes behind $a = a_m a_{m-1} \cdots a_0$ as

$$\text{join\_zero}(a,i) = a_m a_{m-1} \cdots a_0 \underbrace{0000\cdots}_{i \text{ zeroes}}$$

Defining mul(a, b):

variable 'X' holds the value to be returned when we reach '0'.

initialising, $X = 0$ when b has no digits.

$i = n$, 'i' is index of 'b'

while $i \leq n$:

$X = \text{add}(X, \text{join\_zero}(a^* b_i, i))$ — ①

now, x has an updated value and $i = i-1$, decreasing by 1.

Stopped when i reached upto '0'

After stopping iteration, we get answer.

'X' is a binary numeral which is multiplication of 'a' and 'b' as binary.

Here, in ① we have $a^* b_i$, product of single bit 'b_i' with a binary numeral 'a'.

And we are given with 'add' function.