Name - Jay Shah
Nikhil Raina

Homework # 8

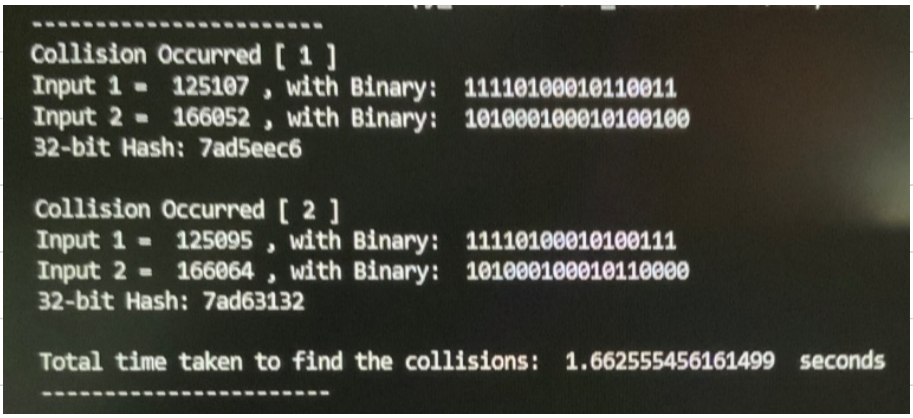
1.

a) The hash function takes an input and then performs the hash that will be explained below:

```
hash (input)
    if input isEmpty:
        return hex(10)    // default value to return if input
                             is empty.
    else:
        x = unicode point of input[0]  // ord(input[0])
        m = 10 // custom power value.
        for c in input:                          → XOR
            x = generate_32_bit((x * m) ^ ord(c))
        x = x ^ length(input)
        if x == -1:
            x = -2
        return hex(absolute(x))[2:]  // avoids '0x'


generate_32_bit (value):
    // converts the value to a 32 bit integer.
    x = value % (2^32)  // to keep the number within
                           2^32 bits.
    if x >= 2^31:
        x = x - 2^32  // same reason as above.
    return int(x)
```

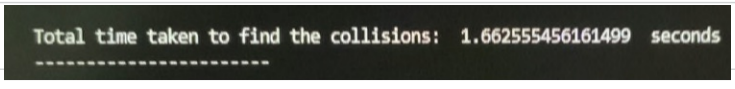
b)



c)

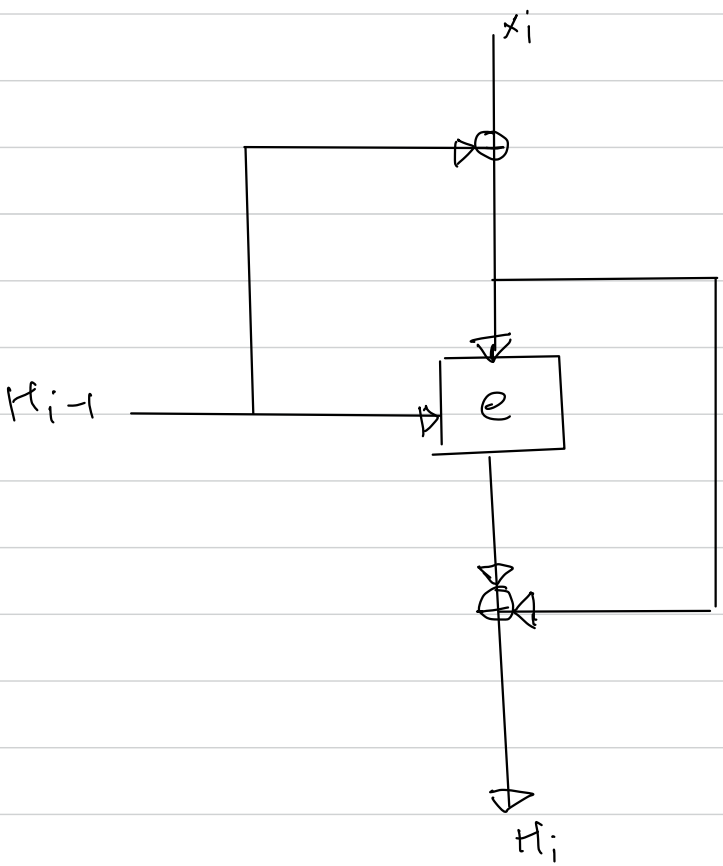
```
collisions = 0
input = 0
hex_values = dictionary()
while True:
    convert input to binary
    hex_resut = hash(converted_binary)
    if hex_resut in hex_values:
        collisions += 1   // collision occurred
    else
        add hex_result as key with the
        converted_binary as the value.
    increment input += 1
```

d)

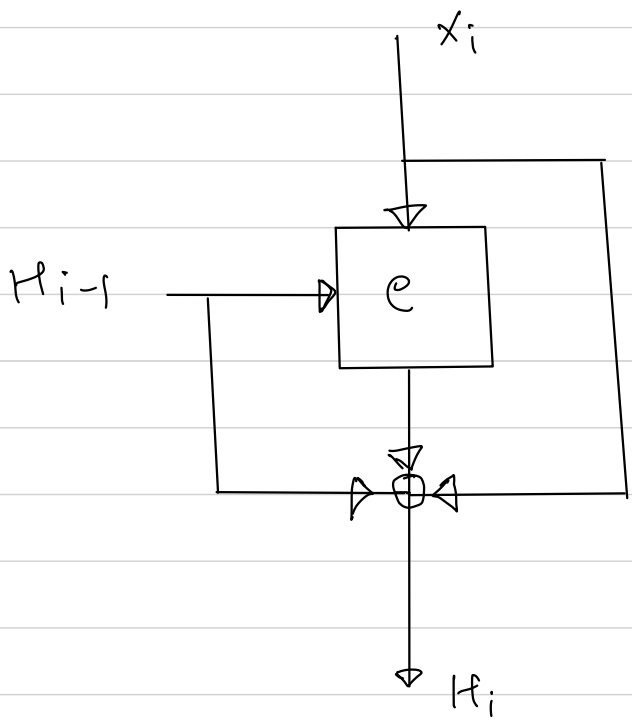


11.3 Draw a block diagram for the Following hash Functions.

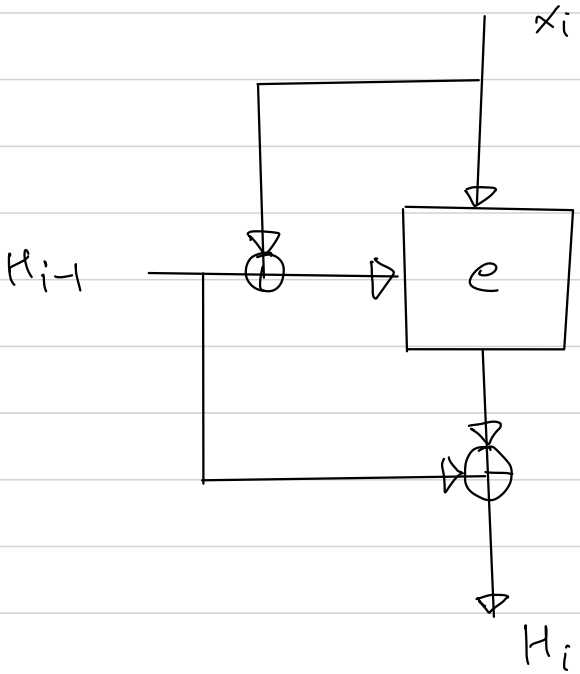
1. $e(H_{i-1}, x_i) \oplus x_i$



2. $e(H_{i-1}, x_i \oplus H_{i-1}) \oplus x_i \oplus H_{i-1}$



3. $E(H_{i-1}, x_i) \oplus x_i \oplus H_{i-1}$



4. $E(x_i \oplus H_{i-1}, x_i) \oplus H_{i-1}$