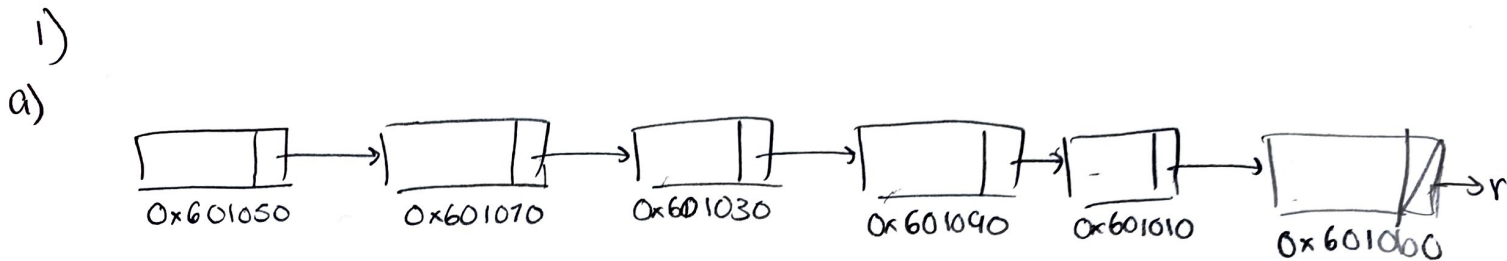


Jonathan Tsao
HW 4
CS 261



b)

yeht	lot	em	d	uuuu
	space	space		null
pmoc	retu	uuu	s	uuuu
		null		null
luoc	no	uuu	yl	uuuu
	space	null		null
a	od	htir	item	uu .e
	space			null
rgw	-	eca	ppoh	uu re
space	space			null

2591

They told me computers could only do
arithmetic. - Grace Hopper 1952

A[R][S][T]

$$A[h][i][j] = x_0 + L(h \cdot \underset{\text{plane}}{[S]} \cdot \underset{\text{row}}{[T]} + i \cdot \underset{\text{col}}{[T]} + j)$$

2.

h stored in rdi/edi
i stored in rsi/esi
j stored in rdx/edx

imulq \$1440, %rsi, %rsi

$$i = i \cdot 1440$$

This shows $i = L(i \cdot [T])$. But we know $L = 24$ bytes.
 $i = 24(i) \cdot [T]$
 $i = i \cdot (24[T])$ $24[T] = 1440$
 $T = 60$

(rdx, rdx, 2), %rax

$$rax = j \cdot 2 + j$$

8(rsp), rdx

When storing variables, we store them in order. Therefore, 8 bytes above rsp is

imulq 86400, %rdi, %rdi

$$h = h \cdot 86400$$

This shows $h = L(h \cdot [S] \cdot [T])$
We know $L = 24$ bytes and $[T] = 60$

$$h = (24)h \cdot [S] \cdot [60]$$

$$86400 = (24)(60)[S]$$

$$S = 60$$

The last known stored value into eax is 2073600, which can correlate to the total size of the 3D node structure. This is because the code says we do:

return sizeof(A) and this returns size in bytes

Thus, A[R][S][T] = 2073600 bytes

Since each node is 24 bytes, we do $2073600 / 24 = 86400$

total nodes. Then, $86400 = [R][S][T] = [R][60][60]$

$$R = 24$$

$$R = 24$$

$$S = 60$$

$$T = 60$$