[illegible] SC   José [illegible] , a100161

4.6)

a) movl $10 , % eax

~~compl~~ compl %ebx, %eax

~~jg~~ jgel .lb2

.lb1: addl %ebx, %ecx
addl %eax, %ecx
jmp .lb~~2~~3

.lb2: subl %eax, %ecx
addl %ebx, %ecx

.lb3: ~~imull~~ addl %ecx %edx
addl %ecx, %edx
addl %~~e~~ecx, %edx

imull $3, %ecx, ~~%~~%edx

b) 4: imull $4 , %ecx, %edx
⋮

7) .globl array (array global)
.size array, 400 (size: 100 x 4 ~~[illegible]~~ ~ sizeof(int))
n posições

.globl sum (variável global)
size sum, 4 (sizeof(sum) = 4, int)
type main, @function (main, função global)

.L5 return 0;
leave

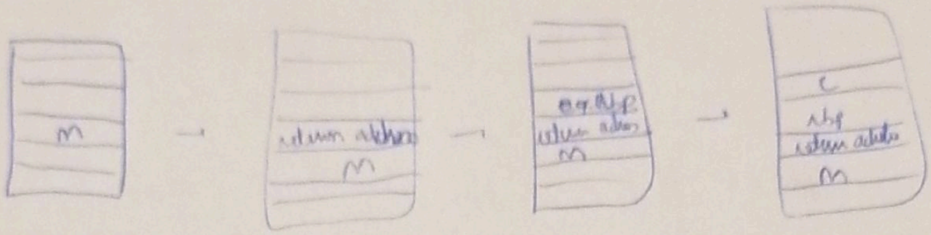
.L3: -4(%rbp) = i
i = 0 (~~[illegible]~~ movl $0, %eax)
~~i~~ scanf(...) (call __isoc99_scanf)
i++; (addl $1, -4(%rbp))

.L2 i - 99 ~~[illegible]~~ <= 0 (cmpl $99, -4(%rbp))
jle .L3
i = 0 (se i-99<=0 não se verifica)
jmp .L4

.L4: a i<100 && array[i] > 0;
jg .L6 , senão jg .L5

.L6: sum += array[i]
i++; (addl $1, -4(%rbp))

4.8)



b) line 5: if (n%2 != 1) ⇒

```
shrl $31, %edx
addl %edx, %eax
andl $1, %eax
subl %edx, %eax
cmpl $1, %eax
```

line 8: badDec2bin (44); ⇒

```
movl $44, %edi
call badDec2bin
```

line 12: for(c=16; c>=0; c--) ⇒

```
movl $16, -4(%rbp)
jmp .L2
```

```
.L2: cmp $0, -4(%rbp)
jns .L5
```

line 13: if(n>>c & 16) ⇒

```
movl -20(%rbp), %edx
movl -4(%rbp), %edx
movl %eax, %ecx
sarl %cl, %edx
movl %edx, %eax
~~%mo~~ andl $16, %eax
testl %eax, %eax
je .L3
```

4.9) (1) 0

(2) y==x

(3) y>x

(4) val=1

(5) val=2

(6)

(7) val=3