

Burak Yesil

I pledge my honor that I have abided by the Stevens Honor System.

Problem 1:

```
MOV X2, 0 //Counter
MOV X3, 0 //Address of Index
MOV X12, 0
```

Loop:

```
    SUBS X4, X2, 12    //checks if X2 is 12
    CBZ X4, Exit       //actually does the conditional

    ADD X10, X19, X12
    LDUR X13, [X10, 0]
    SUB X13, X13, X2
    STUR X13, [X10, 0]
    ADD X2, X2, 1
    LSL X12, X2, 3
    B Loop
```

Exit: ...

Problem 2:

```
ADD X19, X20, X21
LSL X19, X19, 3
ADD X19, X22, X19
LDUR X19, [X19, 0]
STUR X19, [X23, 48]
```

Problem 3:

```
SUBS XZR, X19, X20
B.GE Else
ADD X19, X19, X20
B Exit
```

Else:

```
    LSL X20, X20, 3
    B Exit
```

Exit:

Problem 4:

$36 > 20$

$X1 = 9$

The program starts off by comparing $x0$ and the number 20. Then it checks if $X0$ is greater than or equal to 20. If $X0$ is greater than or equal to 20, the Program branches to the else statement. If this is not the case, the program is done.

Since $36 > 20$, we execute the else statement and divide by 4. This is because we are right shifting by 2.

Problem 5:

The number in register $X10$ is 16. The initial value of $X10$ is 1 and the initial value of $X11$ is also 1. After the branch labeled Loop, the program compares $X11$ and 5 and checks if $X11 > 5$. If it is the case that $X11 > 5$, then it branches to done, otherwise, it adds 1 to $X11$ and multiples $X10$ by 2. This continues until $X11 > 5$.

Problem 6:

Caller:

```
SUB SP, SP, 8
STUR X30, [SP, 0]
MOV X0
MOV X1, 3
```

BL addition

```
LDUR X30, [SP, 0]
ADD SP, SP, 8
```

BR X30

addition:

```
ADD X0, X0, X1
```

BR X30