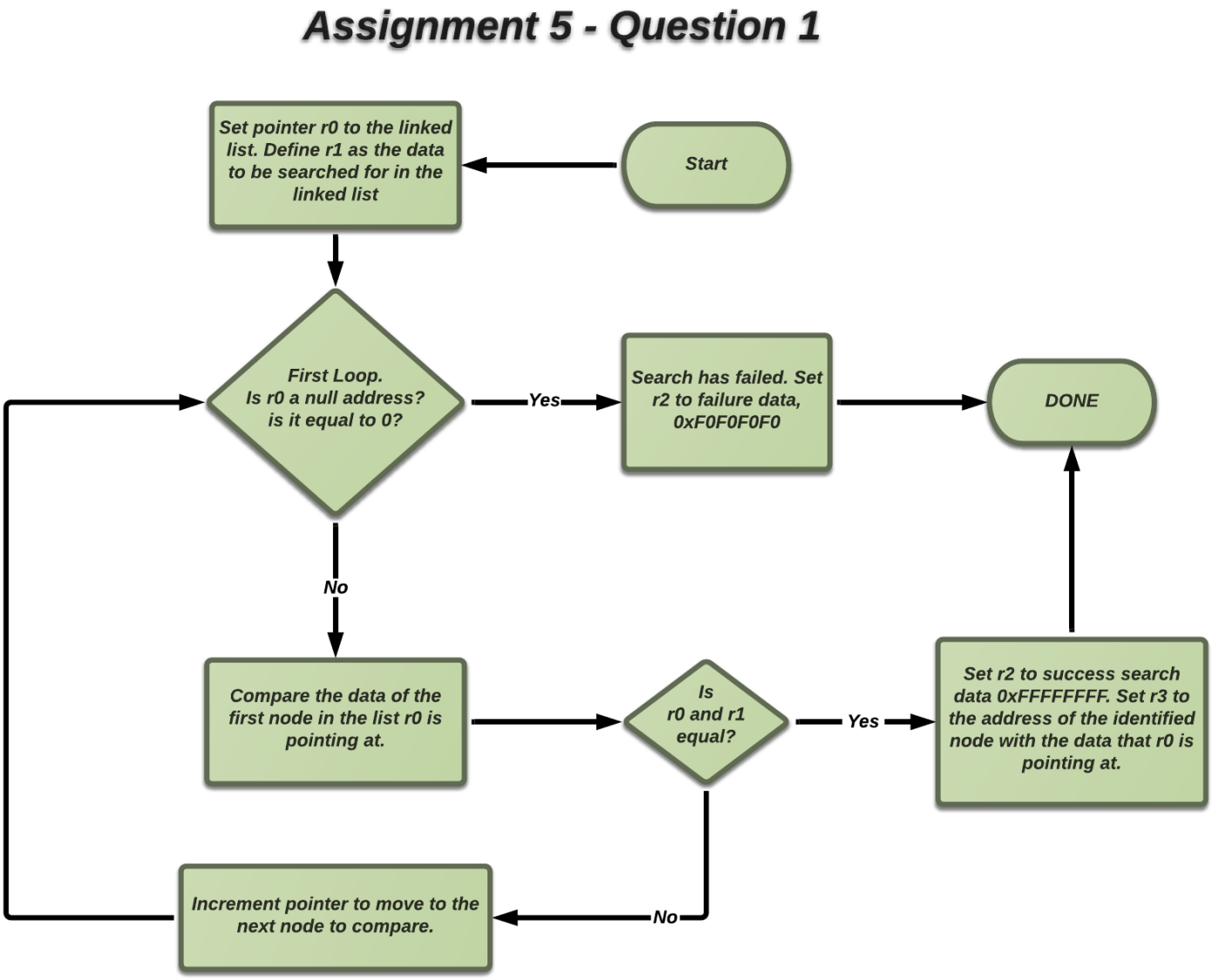
Assignment #5
Josh Jackson - 250722551

Question 1

Flow Chart:

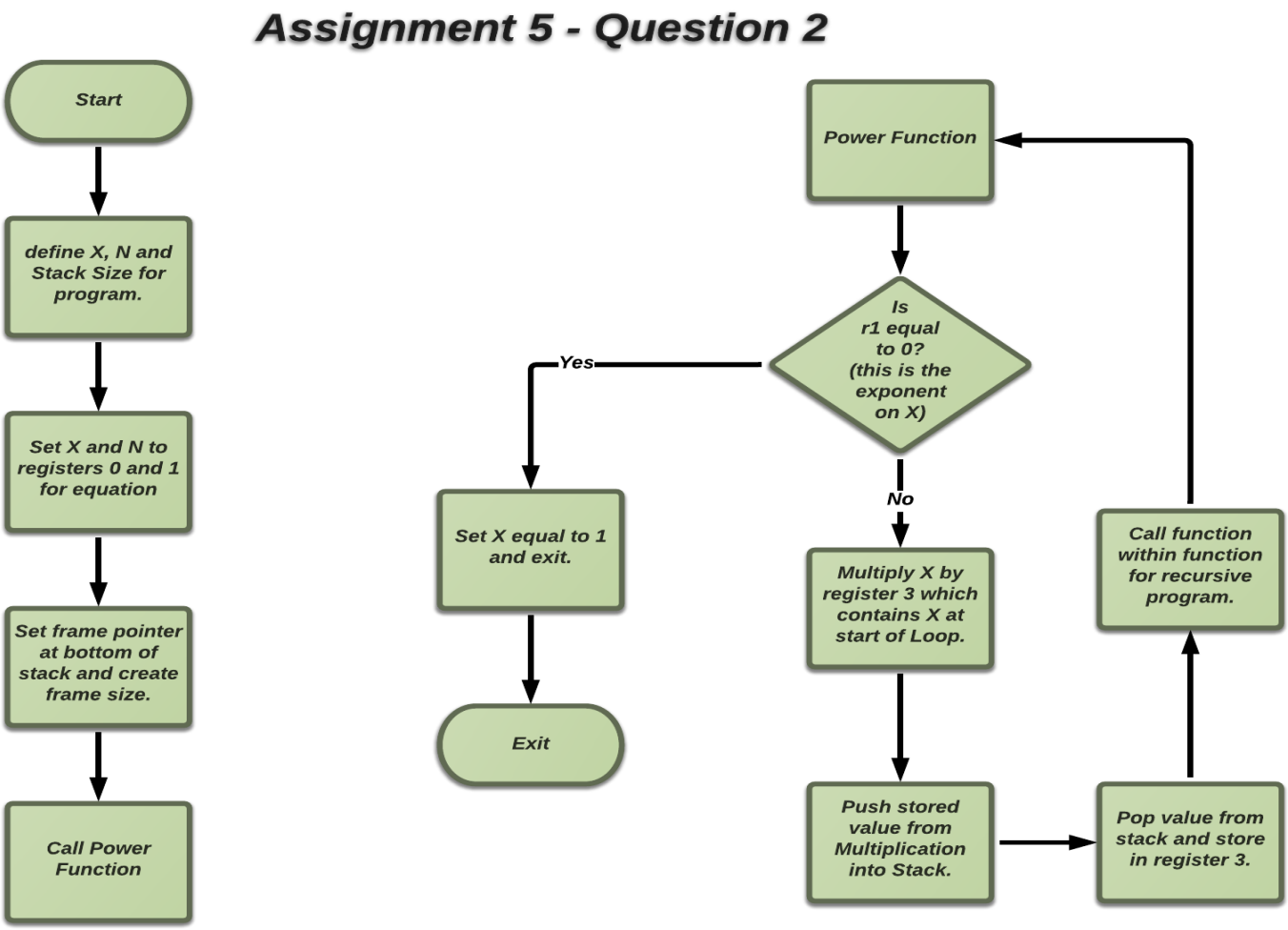


Code:

```
1     AREA a5_question1, CODE, READONLY
2     ENTRY
3
4         LDR     r1,=0x12341111          ;load value into r1 to search for in linked list
5         ADR     r0,List                 ;point r0 to first node of linked list
6
7 Loop    LDR     r4,[r0]                 ;if not equal read the pointer in r
8         CMP     r4,#0                   ;check address of first node not to be null
9 Exit    BEQ     Exit                    ;if reach a null address exit
10        MOVNE   r0,r4                   ;if not null set ro to data of first node in list
11        BNE     Search                  ;if not equal to null enter loop
12
13
14 Search  CMP     r1,r0                   ;compare r1(check val) and r0 (first node)
15        MOVEQ   r2,#0xFFFFFFFF          ;if equal set r2 to success value
16        MOVEQ   r3,r0                   ;and set r3 to address of node where value was found
17        LDRNE   r2,=0xF0F0F0F0          ;if not equal set r2 to failure value
18        BNE     Loop                    ;if not equal take new node and check for null again
19
20 Endless B      Endless                 ;infinite loop
21
22                                        ;data set
23 List   DCD     0x12341111, Item5
24 Item2  DCD     0x12342222, Item3
25 Item3  DCD     0x12343333, Item4
26 Item4  DCD     0x12344444, Item6
27 Item5  DCD     0x12345555, Item2
28 Item6  DCD     0x12346666, Item7
29 Item7  DCD     0x12347777,  0x00               ;terminator
30
31    END
```

Question 2

Flow Chart:



Code:

```
1     AREA a5_question2, CODE, READONLY
2     ENTRY
3
4         MOV     r1,#N               ;set register as exponent value N
5         MOV     r0,#X               ;set register as X value
6         MOV     sp,#stack           ;set stack size
7
8 Main    MOV     r3,#1               ;set r3 to 1 for the first loop through of multiplication
9         BL      power               ;call power fuction
10
11 power  CMP     r1,#0               ;compare exponenet and zero
12        MOVEQ   r0,#1               ;if N is equal to zero make X equal to 1
13        BEQ     exit                ;if equal exit function
14        SUB     r1,r1,#1            ;N minus 1 to decrement exponent count
15        MUL     r2,r0,r3            ;multiply x by itself and store in stack
16        MOV     r3,r2               ;store previous multiplication in r3
17        PUSH    {r2}                ;push value on to stack
18        BL      power               ;recursive call of function within function
19
20 exit   MOV     r3,r2               ;store in r3 which is the variable result
21 Endless B      Endless             ;infinite loop
22
23 stack  EQU     0xFF
24 N      EQU     4
25 X      EQU     2
26 result EQU     r3                  ;result will found in register3
27    END
28
```

How many stack frames are needed to calculate x^n, when n = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and 12?

In order to calculate x^n you would need a stack frame for every exponent after every multiplication. For example, 0 would need 1 frame, 1 would need 2, 3 would need 4, and so on.