```
                              AREA question1, CODE, READONLY

                              ENTRY


                              LDR sp, =BoS
          ;Set the stack pointer to the end of the stack space

                              LDR fp, =BoS
          ;Set the frame pointer to the end of the stack space

                              SUB sp,sp,#12
          ;Move up the stack pointer by 12 for: x,n,and result

                              LDR r0,X
          ;Load value for x into r0

                              LDR r1,N
          ;Load value for N into r1

                              STMDB fp, {r0,r1}
          ;Store the value onto the stack for function access


                    ;fp -12 is left for return value

                              BL func
          ;Function call

                              LDR r2,[sp]


                              ADD sp,sp,#12

                              STR r2,result

finish              B finish



;---------------------------------------

func                SUB sp,sp,#16                                                                    ;Move
up the stack pointer by 16 for: x,n,fp and sp

                              STMIA sp, {r0,r1,fp,lr}                                                ;Store
all value onto the stack

                              LDMDB fp, {r0,r1}
          ;Call by value of X and N
```

```
                        MOV fp,sp
        ;Line up fp and sp
                        SUB sp,sp,#12
        ;Move up the stack pointer by 12 for: x,n,and result


                        CMP r1, #0
        ;Base case: N=0
                        MOVEQ r0, #1
        ;The return value is stored in r0
                        BEQ return



                        TST r1, #1
        ;Test is N is even
                        BEQ     even
        ;If it is odd:
                        SUB r1, #1
        ;N is subtracted by one, ready for the next funtion call
                        STMDB fp, {r0,r1}
        ;The value of X and new N is store onto the stack
                        BL func
        ;Recursive call for updated N
                        LDR r1,[sp]
        ;When result is returned, it is loaded into r1
                        MUL r0,r1,r0
        ;The current x will be multiplied by the return result for the recursive call
                        B return
        ;The result will be returned


even            LSR r1, #1
        ;If N is even: It is first halfed
                        STMDB fp, {r0,r1}
        ;Then the new value is stored and ready for new funtion call
```

```
                    BL func
        ;Recursive call with new N

                    LDR r1,[sp]
        ;The return value of the recursive call is stored in r1

                    MUL r0,r1,r1
        ;It is squared and stored in r0

                    B return
        ;The value is returned


return          STR r0, [fp, #16]                                   ;The value in r0
will be the return value

                    ADD sp,sp,#28
        ;The stack will be collapsed by adding 28 (12+16) to the stack pointer

                    LDMIA fp, {r0,r1,fp,pc}                                 ;The
original value will be restored


;--------------------------------------------------------------

                    AREA question2, DATA, READONLY

X                   DCD         3
                ;Value for X

N                   DCD     5
        ;Value for N

result          SPACE   4
        ;Left space for storing the result

storage         SPACE   200
        ;The space of the stack needed

BoS                 DCD     0x00
        ;Address of bottom of stack


                    END
```

Stack structure

sp
Fp
N(new)
X
Result
N
X