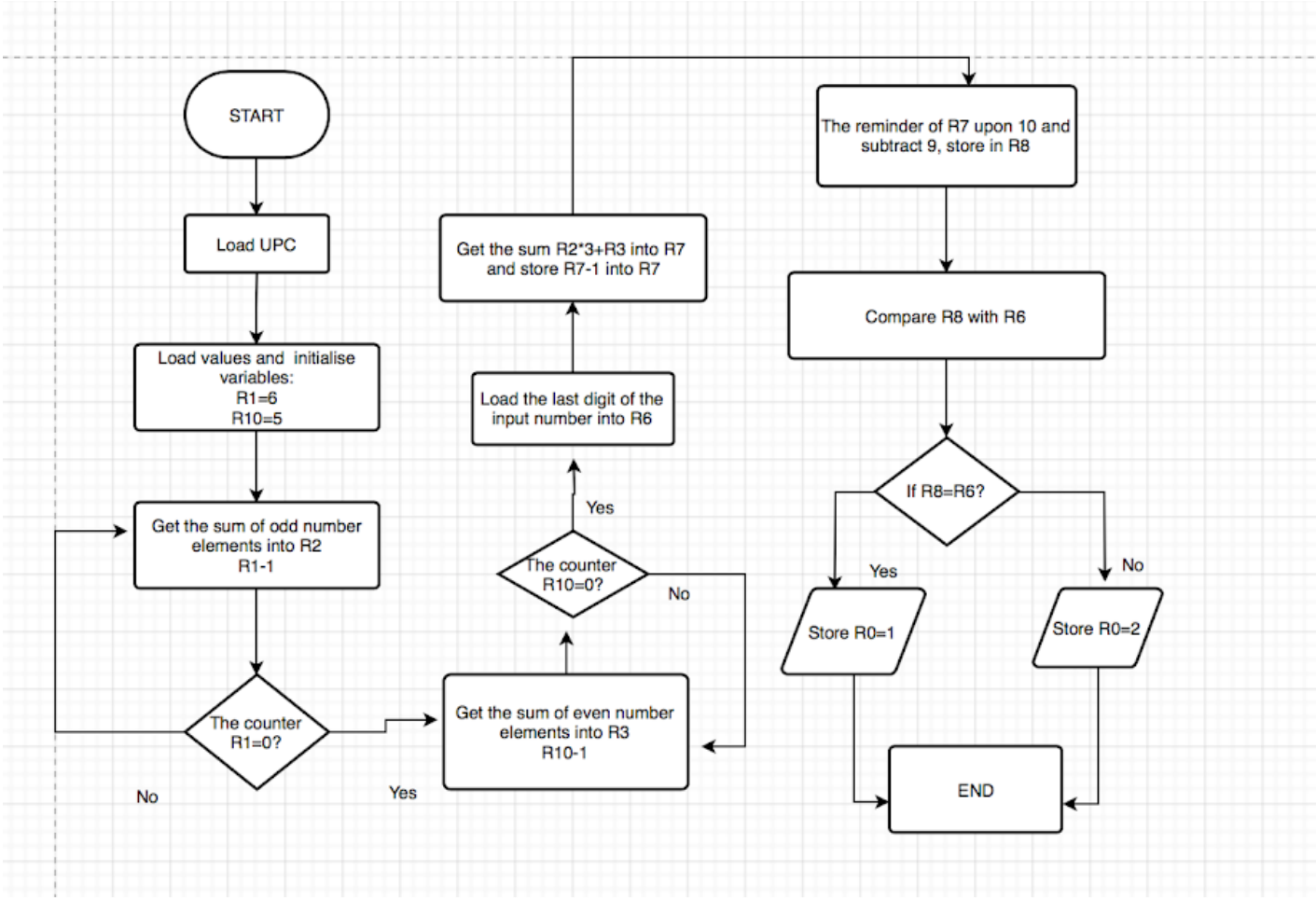
Student number:250847432

Student name: Junlin pang

Part 1 flow chart



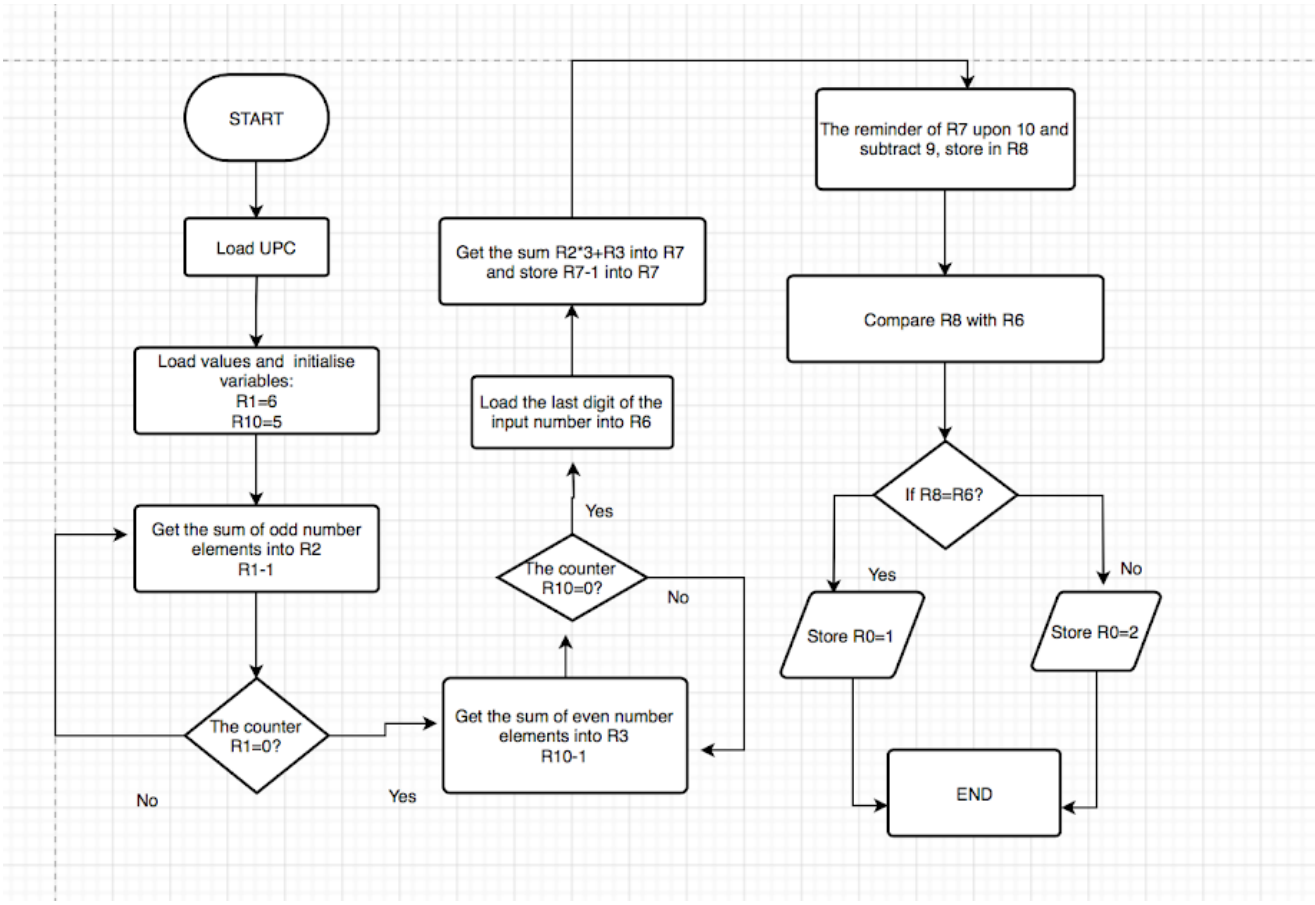
```
        AREA assignment1, CODE, READONLY
        ENTRY
        ADR R12,UPC;
        ADR R11,UPC+1;
        MOV R1, #6                  ;initialize the counter for sum of odd position
into 6
        MOV R10, #5;            initialize teh counter for sum of even position into 5
        MOV R2, #0;                 clear the sum in r2 (for odd)
        MOV R3, #0;                 clear the sum in r3 (for even)
Loop
        LDRB R4, [R12];     Load the first element pointed by R12 into R4
        ADD R12, R12,#2;        Point to the next odd element in the series.
        SUB R4, R4,#0X30;       change the character into the exact number
        ADD R2, R2,R4;          Add to the total sum of odd element to R2
        SUBS R1, R1,#1;         Decrement to the loop counter by 1.
        CMP R1,#0;                  compare the counter with 1
        BNE Loop;
Loop1
        LDRB R5, [R11];         load the element after R0 into R5
        ADD R11, R11,#2;        point to the next even element
        SUB R5, R5,#0x30;       change teh character into integer
        ADD R3, R3, R5;         get the sum of even element
        SUBS R10, R10,#1;       decrease the even counter
        CMP R10,#0;                 compare the counter with 1
        BNE Loop1;
```

```
                B NEXT;

NEXT
                LDRB R6, UPC+11 ;       Get the last element???
                SUB R6, R6,#0x30;       change the character into integer
                MOV R9, #3;
                MUL R9, R2, R9;         multiply by 3
                ADD R7, R9, R3;         get the total sum
                SUB R7,R7,#1;           minus 1

MOD10
                SUB R7, R7, #10;
                CMP R7, #10;            check if r7 is less than 10
                BPL MOD10;                      repeat the mod
                RSB R8, R7, #9;
                CMP R8, R6;                     compare R8 with R6
                BEQ Valid;
                BNE Invalid;
Valid
                MOV R0,#1;
                B Exit
Invalid
                MOV R0,#2;
Exit    B Exit
        ;ALIGN
                AREA assignment1, DATA, READWRITE
UPC     DCB "013800150738" ;  #UPC string
                END
```

Part2 flow chart




```
                AREA assignment2,CODE, READWRITE
                ENTRY
                ADR R1, STRING;          put string into R1
                MOV R2, #0x00;           initialize R2
                MOV R7, #0;              initialize counter R7
COUNTER         LDRB R3,[R1,R7];         load the element into R3 from r1 pointed by r7
                ADD R7,R7,#1;            increase the counter
                CMP R2, R3;              get the number of whole elments
```

```
                BNE COUNTER;            loop
                SUB R7,#1;
                ADD R8,R1,R7;
                BEQ LOOP;                   if equal then goes to get the first
element

LOOP            LDRB R3,[R1];           get the first element
                ADD R1,R1,#1;           increase the counter
                CMP R3, #0x41;          check if bigger than A
                BLT LOOP;                   else goes to get next element
                CMP R3, #0x5A;          ccheck if more than Z
                BLT NEXT;                   go to get the last element if is between
A to Z
                CMP R3, #0x61;          check if more than a
                BLT LOOP;                   else go to get the next element
                CMP R3, #0x7A;          check if smaller than z
                BLT CHANGING;               if between a and z, change it into
capital letter
                BGT LOOP;                   else go to get another element.

CHANGING        SUB R3,R3,#0x20;        change into capital letter
                B NEXT

NEXT            SUB R8,#1;                  decrease the counter
                LDRB R5,[R8];           get the last element
                CMP R5, #0x41;              find the letter same with previous
                BLT NEXT
                CMP R5, #0x5A
                BLT THEN
                CMP R5, #0x61
```

```
                BLT NEXT
                CMP R5, #0x7A
                BLT CHANGING2
                BGT NEXT

CHANGING2   SUB R5,R5,#0x20;        change into capital letter
                B THEN
THEN        CMP R3,R5;                  compare the R3 and r5
                BNE NOTP;                   if not same, then it's not palindrom
                CMP R8, #0xBC;          if same, compare R8 with 0xBC
                BEQ PALINDROM;              if equal, then it's palindrom
                BNE LOOP;                   if not, back to loop get the next
element
PALINDROM   MOV R0, #1
                B EXIT
NOTP        MOV R0, #2
            B EXIT

EXIT        B EXIT
                AREA assignment2, DATA, READWRITE
STRING      DCB "He lived as a devil, eh?"                  ;string
EoS         DCB 0x00
      ;end of string
                END
```