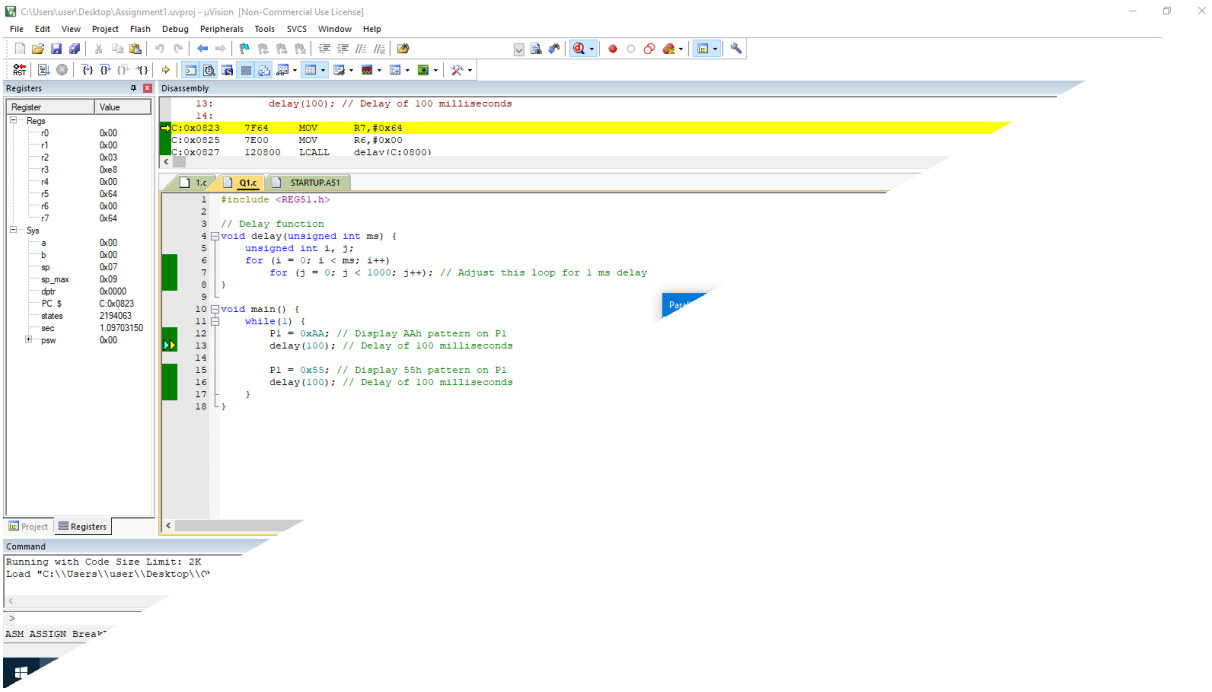
Assignment 6

➢ Write Program in KEIL Embedded C:

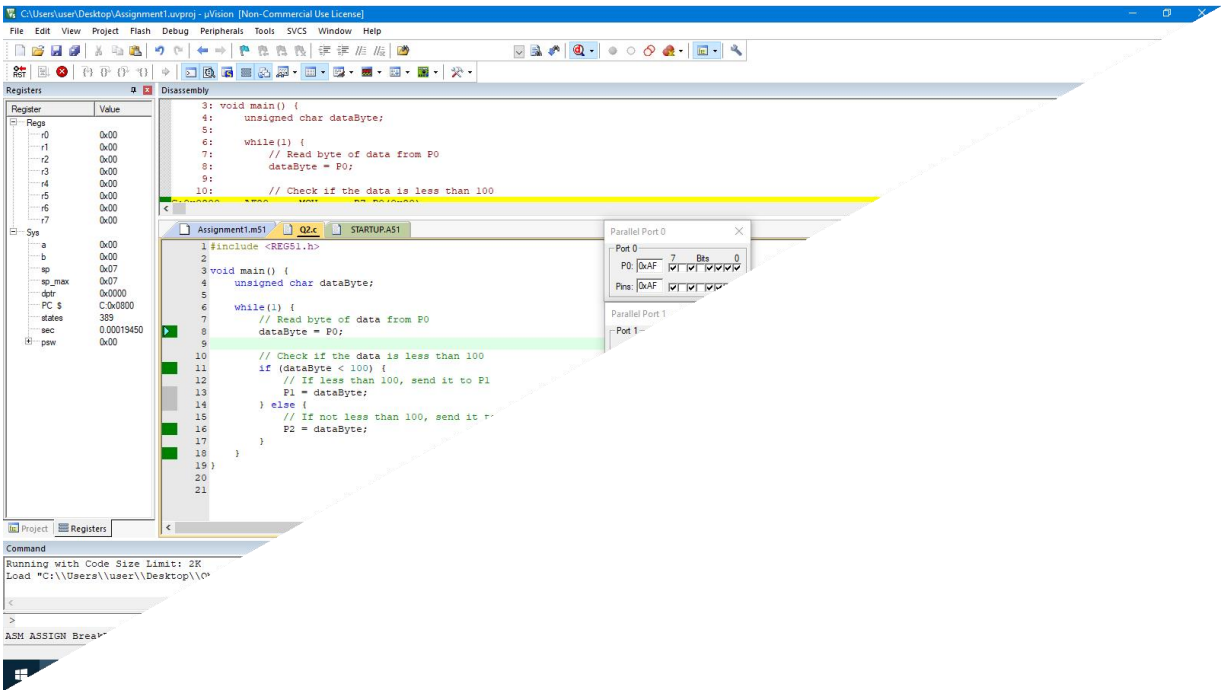
1. Write a C program for the 8051 to display a pattern of AA and 55h on port P1 with the delay of 100 ms.

```
#include <REG51.h>
// Delay function
void delay(unsigned int ms) {
unsigned int i, j;
for (i = 0; i < ms; i++)
for (j = 0; j < 1000; j++); // Adjust this loop for 1 ms delay
}
void main() {
while(1) {
P1 = 0xAA; // Display AAh pattern on P1
delay(100); // Delay of 100 milliseconds
P1 = 0x55; // Display 55h pattern on P1
delay(100); // Delay of 100 milliseconds
}
}
```



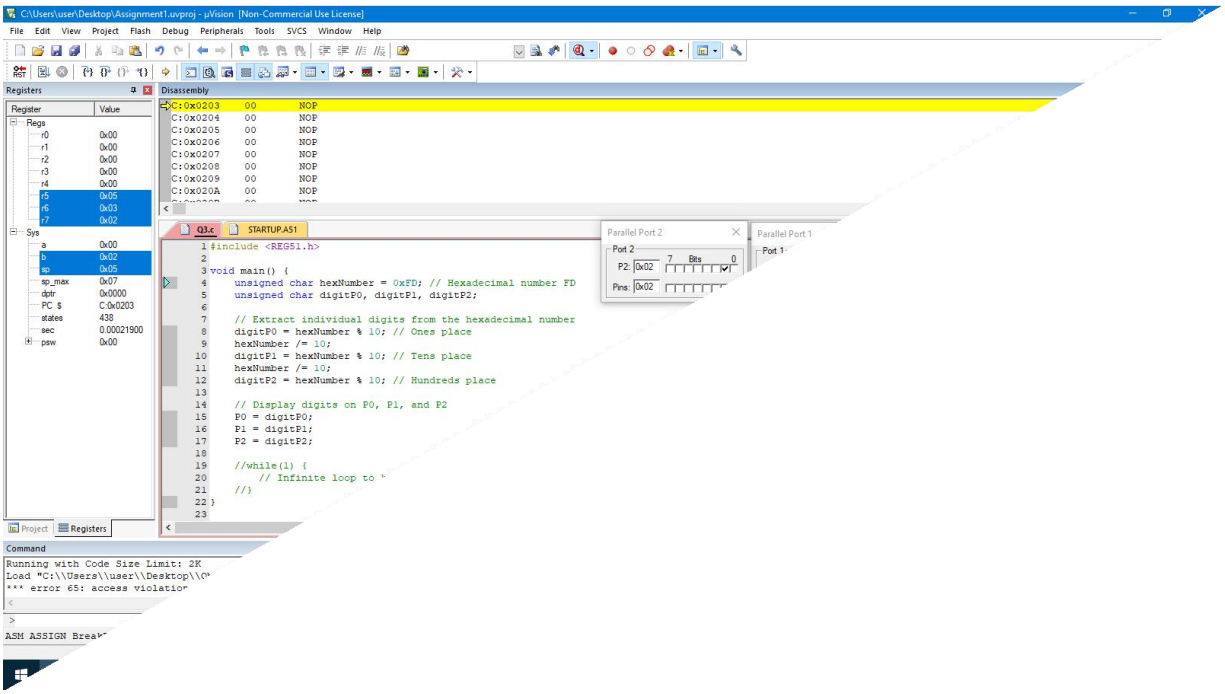
2. Write an 8051 C program to get a byte of data from P0. If it is less than 100, send it to P1; otherwise, send it to P2.

```
#include <REG51.h>
void main() {
unsigned char dataByte;
while(1) {
// Read byte of data from P0
dataByte = P0;
// Check if the data is less than 100
if (dataByte < 100) {
// If less than 100, send it to P1
P1 = dataByte;
} else {
// If not less than 100, send it to P2
P2 = dataByte;
}
}
}
```



3. Write an 8051 C program to convert 11111101 (FD hex) to decimal and display the digits on P0, P1 and P2.

```
#include <REG51.h>
void main() {
unsigned char hexNumber = 0xFD; // Hexadecimal number FD
unsigned char digitP0, digitP1, digitP2;
// Extract individual digits from the hexadecimal number
digitP0 = hexNumber % 10; // Ones place
hexNumber /= 10;
digitP1 = hexNumber % 10; // Tens place
hexNumber /= 10;
digitP2 = hexNumber % 10; // Hundreds place
// Display digits on P0, P1, and P2
P0 = digitP0;
P1 = digitP1;
P2 = digitP2;
}
```



➢ To be done using EdSim51 simulator in 8051:

1. Write a Program to check whether a number is palindrome or not. If palindrome store FFh in accumulator.

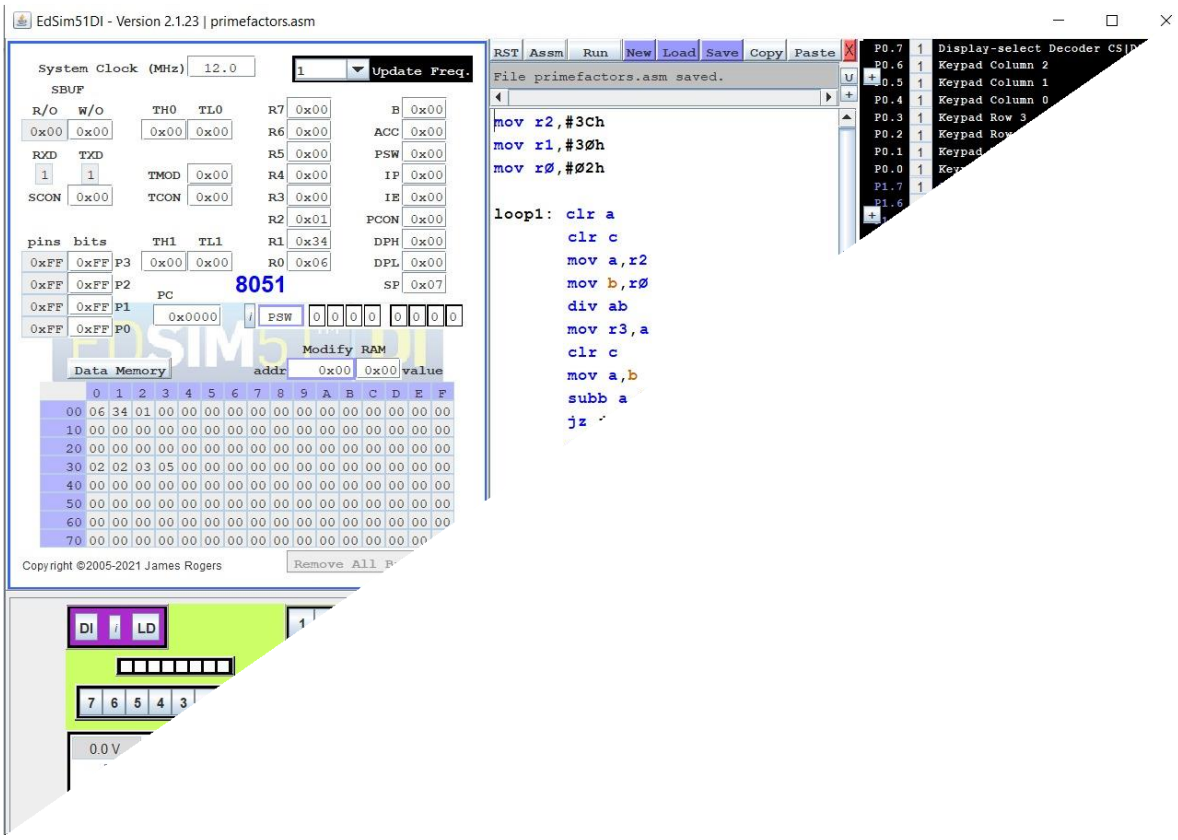
```
ORG 0000H
MOV DPTR,#8000H
MOVX A,@DPTR
MOV R0, A
MOV R2, #00H
BACK: MOV B, #0AH
DIV AB
MOV B, A
MUL AB
SUBB A, R0
JNZ NOT_PALINDROME
INC R2
JMP NEXT
NOT_PALINDROME: CLR A
JMP  STORE_RESULT
NEXT: INC DPTR
MOVX A, @DPTR
CJNE A, #0FFH, BACK
JMP STORE_RESULT
```

```
STORE_RESULT: MOV A, #0FFH
MOV DPTR, #8100H
MOVX @DPTR, A
END
```

2. Write an assembly language program to compute prime factors of a number.

```
mov r2,#3Ch
mov r1,#30h
mov r0,#02h
loop1: clr a
clr c
mov a,r2
mov b,r0
div ab
mov r3,a
clr  c
mov a,b
subb a,#00h
jz join
clr   a
clr   c
inc r0
mov a,r3
```

```
subb a,#00h
jz exit
sjmp loop1
join: mov a,r0
mov @r1,a
inc r1
mov a,r3
mov r2,a
sjmp loop1
exit: nop
end
```



3. Write an assembly language program to print Binary Pattern on the Port 1

```
ORG 0000H
START: MOV P1, #00000000B
LOOP: MOV A, P1
CPL A
MOV P1, A
CALL DELAY
```

```
SJMP LOOP
DELAY: MOV R2, #0FFH
L1: DJNZ R2, L1
RET
END
```