Name:SOUNG VICHKA

ID:e20240672

Group:I2-GIC2-B

TP12

Computer Architecture (Assembly Language smz32v50)

TP 5: Create a very simple calculator

The calculator is very simple which can only operate addition and substraction without negative value and the result are between 0-99.

- ✓ 2 + 3 is Ok
- ✓ 20+ 7 is Ok
- ✓ 80 + 22 is not Ok as it is exceed 99
- ✓ 3 - 4 is not Ok as it give -1 which is negative value
- ✓ -1 + 4 is not Ok as the input is negative

You don't need to detect/prevent the above condition, just consider that consider that user won't violate this rule.

The process flow:

1. User run program
2. Program ask user to input number
3. User type in a positive number, and program display it in 7 segments display
4. Program ask user to input again for the operator (addition or substruction)
5. User type in + or -
6. Program ask user to input again for the 2nd number
7. User type in a positive number, and program display it in 7 segments display (clear the previous value)
8. Program display result in 7 segment display
9. type any key, program start again. 7 segments display is cleared.

```
Code: jmp init

db FB ; 0
db 0B ; 1
db B7 ; 2
db 9F
db 4F
db DD
db FD
db 8B
db FF
db DF ; 9 is at address B
db "+-*/"

init:
mov dl, 0
mov cl, c0

input:
in 00
sub al, 30
push al

inc dl
cmp dl, 2
jz input

in 00
sub al, 30
pop bl
mul bl ,A
add al, bl


display:
mov [cl], al
div al, A
add al, 02
mov al, [al]
dec al
out 02

mov al, [cl]
mod al, A
add al, 02
mov al, [al]
out 02

inc cl
cmp cl, c1
jz input
cmp cl, c3
jz stop


check_operator:
mov al, [c0]
```

```
mov bl, [c1]

pop dl           ; dl = +
add dl, 30

cmp dl, [C]
jz plus

cmp dl, [D]
jz subtract

cmp dl, [E]
jz multiply

jmp division

plus:
add al, bl
jmp display

subtract:
sub al, bl
jmp display

multiply:
mul al, bl
jmp display

division:
div al, bl
jmp display


stop:
end
```