- increment R0
  STR current Add

  0 1 2 3

  left char R1
  right char

  R5

  left Addr
  Right Addr

  Stop when left ≥ right

- Find Minimum Index :

  - ptr to min element : R0 (initially 1st)
  - min element's value : R1
  - ptr to each element : R2
  - each element's value : R3
  - - addr of final element : R4

  : 1 2 3 4

- caesar cipher :

  b
  a
  mod_return
  String
  Shift

  - caesar :
    find length
    get arguments for mod and mod_return
    (push a, push b)

  - mod :
    R4 := a (pop)

R5 := 6 (pop)

Work on R4, R5 → mod_return

push (mod_return)

ret (caesar)

(mod_return
  String
  Shift)

) Leapfrog :

R0 : inc

R1 : func_addr

R2 : value at func_addr

R3 : copy_addr = func_addr + inc

R4 : start_copy = copy_addr

R5 : stop_addr

negate stop_addr

if (R4 + R5 == 0)

HALT

while R2 + R2 != 0

restore arr

LDR R2, R3, #0

R1++

R3++

R2 = 0

STR R3, R2

JMP R4

-R3 -R4

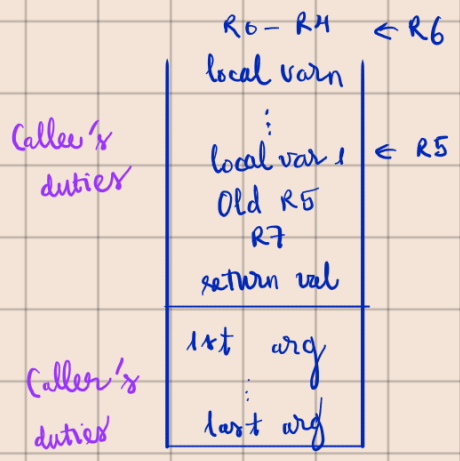
·) merge sort

MERGESORT (arr, buf, start, end)

MERGE(arr , buf , start , mid , end) 5 , 2 , 3 , 1

1st run : 16384 , 20480 , 0 , 1 , 2

i = 0 j = 1 , k = 0



start 0

mid 1

end 2

1
1
1

i 0

j 1

R0