Breakpoint 1, \_continue\_loop () at Lab8\_1.s:11

11 BLE \_loop

(gdb) info r

r0 0x0 0

r1 0x1 1

r2 0xbefff37c 3204445052

r3 0x103d0 66512

r4 0x0 0

r5 0x103ec 66540

r6 0x102e0 66272

r7 0x0 0

r8 0x0 0

r9 0x0 0

r10 0xb6fff000 3070226432

r11 0x0 0

r12 0xbefff2a0 3204444832

sp 0xbefff228 0xbefff228

lr 0xb6e6a718 -1226397928

pc 0x103e4 0x103e4 <\_continue\_loop+4>

cpsr 0x80000010 -2147483632

fpscr 0x0 0

Breakpoint 1, \_continue\_loop () at Lab8\_1.s:11

11 BLE \_loop

(gdb) info r

r0 0x1 1

r1 0x1 1

r2 0xbefff37c 3204445052

r3 0x103d0 66512

r4 0x0 0

r5 0x103ec 66540

r6 0x102e0 66272

r7 0x0 0

r8 0x0 0

r9 0x0 0

r10 0xb6fff000 3070226432

r11 0x0 0

r12 0xbefff2a0 3204444832

sp 0xbefff228 0xbefff228

lr 0xb6e6a718 -1226397928

pc 0x103e4 0x103e4 <\_continue\_loop+4>

cpsr 0x80000010 -2147483632

fpscr 0x0 0

(gdb) disassemble \_loop

Dump of assembler code for function \_loop:

0x000103dc <+0>: add r0, r0, r1

End of assembler dump.

(gdb) x/10i main

0x103d0 <main>: mov r0, #0

0x103d4 <main+4>: mov r1, #1

0x103d8 <main+8>: b 0x103e0 <\_continue\_loop>

0x103dc <\_loop>: add r0, r0, r1

0x103e0 <\_continue\_loop>: cmp r0, #9

=> 0x103e4 <\_continue\_loop+4>: ble 0x103dc <\_loop>

0x103e8 <end>: bx lr

0x103ec <\_\_libc\_csu\_init>: push {r4, r5, r6, r7, r8, r9, r10, lr}

0x103f0 <\_\_libc\_csu\_init+4>: mov r7, r0

0x103f4 <\_\_libc\_csu\_init+8>:

ldr r6, [pc, #72] ; 0x10444 <\_\_libc\_csu\_init+88>

(gdb) i b

Num Type Disp Enb Address What

1 breakpoint keep y 0x000103e4 Lab8\_1.s:11

breakpoint already hit 3 times

จงนําโปรแกรมภาษาแอสเซมบลีสําหรับคํานวณค่า mod ในการทดลองที่ 7 มาเรียกใช้ผ่านโปรแกรม

ภาษา C (maybe find ways to do other case than +%+ as well)