

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

Лабораторна робота №4

Системні виклики та асемблер

Прийняв:

Викладач

Ковтунець О.В.

«__»_____2023 р.

Виконав:

Студент 3 курсу, гр. ІК-03

Смірнов Андрій

Олександрович

2023 р.

Мета роботи

Ознайомлення з мовою програмування Асемблер, використання її для вирішення задач управління ОС, засвоєння навичок написання програм з використанням цієї мови програмування.

Загальні відомості до виконання роботи

Результатом виконання роботи є програма для ОС Linux, написана мовою програмування Асемблер (з використанням GNU Assembler або Netwide Assembler), яка опрацьовує надані текстові файли (логи)

- файл 01-tiny-access.log;
- файл 05-huge-access.log;

у форматі ACCESS LOG веб-сервера APACHE і демонструє відповідну вашому варіантові інформацію – аналогічно програмі, написаній мовою програмування C у л.р. №3.

Хід роботи

Транспільовано з коду C за допомогою компілятора gcc.

Код Асемблеру:

```
.section    __TEXT,__text,regular,pure_instructions
.build_version macos, 12, 0    sdk_version 12, 3
.globl _updateCount            ; -- Begin function updateCount
.p2align    2
_updateCount:                  ; @updateCount
    .cfi_startproc
; %bb.0:
    sub     sp, sp, #48
    stp     x29, x30, [sp, #32]    ; 16-byte Folded Spill
    add     x29, sp, #32
    .cfi_def_cfa w29, 16
```

```

.cfi_offset w30, -8
.cfi_offset w29, -16
stur    x0, [x29, #-8]
stur    w1, [x29, #-12]
str      x2, [sp, #8]
str      wzr, [sp, #4]
b        LBB0_1

```

LBB0_1: ; =>This Inner Loop Header: Depth=1

```

ldr      w8, [sp, #4]
ldur     w9, [x29, #-12]
subs     w8, w8, w9
b.ge     LBB0_6
b        LBB0_2

```

LBB0_2: ; in Loop: Header=BB0_1 Depth=1

```

ldur     x8, [x29, #-8]
ldrsw    x9, [sp, #4]
add       x0, x8, x9, lsl #4
ldr       x1, [sp, #8]
bl        _strcmp
cbnz     w0, LBB0_4
b        LBB0_3

```

LBB0_3:

```

ldur     x8, [x29, #-8]
ldrsw    x9, [sp, #4]
add       x9, x8, x9, lsl #4
ldr       w8, [x9, #12]
add       w8, w8, #1
str       w8, [x9, #12]
b        LBB0_7

```

LBB0_4: ; in Loop: Header=BB0_1 Depth=1

b LBB0_5

LBB0_5: ; in Loop: Header=BB0_1 Depth=1

ldr w8, [sp, #4]

add w8, w8, #1

str w8, [sp, #4]

b LBB0_1

LBB0_6:

ldur x8, [x29, #-8]

ldursw x9, [x29, #-12]

add x0, x8, x9, lsl #4

ldr x1, [sp, #8]

mov x2, #11

bl ____strcpy_chk

ldur x8, [x29, #-8]

ldursw x9, [x29, #-12]

add x9, x8, x9, lsl #4

mov w8, #1

str w8, [x9, #12]

b LBB0_7

LBB0_7:

ldp x29, x30, [sp, #32] ; 16-byte Folded Reload

add sp, sp, #48

ret

.cfi_endproc

; -- End function

.globl _compareEntries ; -- Begin function compareEntries

.p2align 2

_compareEntries: ; @compareEntries

```

        .cfi_startproc
; %bb.0:
        sub    sp, sp, #32
        .cfi_def_cfa_offset 32
        str    x0, [sp, #24]
        str    x1, [sp, #16]
        ldr    x8, [sp, #24]
        ldr    w8, [x8, #12]
        str    w8, [sp, #12]
        ldr    x8, [sp, #16]
        ldr    w8, [x8, #12]
        str    w8, [sp, #8]
        ldr    w8, [sp, #8]
        ldr    w9, [sp, #12]
        subs   w0, w8, w9
        add    sp, sp, #32
        ret
        .cfi_endproc

                                ; -- End function

.section    __TEXT,__literal8,8byte_literals
.p2align    3                                ; -- Begin function printTopDates
lCPI2_0:
        .quad 0x4059000000000000            ; double 100
        .section    __TEXT,__text,regular,pure_instructions
        .globl _printTopDates
        .p2align    2
_printTopDates:                                ; @printTopDates
        .cfi_startproc
; %bb.0:

```

```

sub    sp, sp, #96
stp    x29, x30, [sp, #80]      ; 16-byte Folded Spill
add    x29, sp, #80
.cfi_def_cfa w29, 16
.cfi_offset w30, -8
.cfi_offset w29, -16
stur   x0, [x29, #-8]
stur   w1, [x29, #-12]
stur   wzr, [x29, #-20]
stur   wzr, [x29, #-16]
b      LBB2_1

```

LBB2_1: ; =>This Inner Loop Header: Depth=1

```

ldur   w8, [x29, #-16]
ldur   w9, [x29, #-12]
subs   w8, w8, w9
b.ge   LBB2_4
b      LBB2_2

```

LBB2_2: ; in Loop: Header=BB2_1 Depth=1

```

ldur   x8, [x29, #-8]
ldursw      x9, [x29, #-16]
add    x8, x8, x9, lsl #4
ldr    w9, [x8, #12]
ldur   w8, [x29, #-20]
add    w8, w8, w9
stur   w8, [x29, #-20]
b      LBB2_3

```

LBB2_3: ; in Loop: Header=BB2_1 Depth=1

```

ldur   w8, [x29, #-16]
add    w8, w8, #1

```

stur w8, [x29, #-16]

b LBB2_1

LBB2_4:

adrp x0, l_.str@PAGE

add x0, x0, l_.str@PAGEOFF

bl _printf

stur wzr, [x29, #-16]

b LBB2_5

LBB2_5: ; =>This Inner Loop Header: Depth=1

ldur w9, [x29, #-16]

ldur w10, [x29, #-12]

mov w8, #0

subs w9, w9, w10

stur w8, [x29, #-36] ; 4-byte Folded Spill

b.ge LBB2_7

b LBB2_6

LBB2_6: ; in Loop: Header=BB2_5 Depth=1

ldur w8, [x29, #-16]

subs w8, w8, #10

cset w8, lt

stur w8, [x29, #-36] ; 4-byte Folded Spill

b LBB2_7

LBB2_7: ; in Loop: Header=BB2_5 Depth=1

ldur w8, [x29, #-36] ; 4-byte Folded Reload

tbz w8, #0, LBB2_10

b LBB2_8

LBB2_8: ; in Loop: Header=BB2_5 Depth=1

ldur x8, [x29, #-8]

ldursw x9, [x29, #-16]

```
add    x8, x8, x9, lsl #4
ldr     s1, [x8, #12]
; implicit-def: $d0
mov.16b   v0, v1
sshll.2d   v0, v0, #0
; kill: def $d0 killed $d0 killed $q0
scvtf   d0, d0
ldur     s2, [x29, #-20]
; implicit-def: $d1
mov.16b   v1, v2
sshll.2d   v1, v1, #0
; kill: def $d1 killed $d1 killed $q1
scvtf   d1, d1
fdiv     d0, d0, d1
adrp     x8, ICPI2_0@PAGE
ldr      d1, [x8, ICPI2_0@PAGEOFF]
fmul     d0, d0, d1
stur     d0, [x29, #-32]
ldur     w8, [x29, #-16]
add      w12, w8, #1
ldur     x8, [x29, #-8]
ldursw    x9, [x29, #-16]
add      x10, x8, x9, lsl #4
ldur     x8, [x29, #-8]
ldursw    x9, [x29, #-16]
add      x8, x8, x9, lsl #4
ldr      w8, [x8, #12]
; implicit-def: $x9
mov     x9, x8
```



```

ldur    d0, [x29, #-32]
adrp    x0, l_.str.1@PAGE
add     x0, x0, l_.str.1@PAGEOFF
mov     x8, sp

                                ; implicit-def: $x11
mov     x11, x12
str     x11, [x8]
str     x10, [x8, #8]
str     x9, [x8, #16]
str     d0, [x8, #24]
bl      _printf
b       LBB2_9
LBB2_9:                                ; in Loop: Header=BB2_5 Depth=1
ldur    w8, [x29, #-16]
add     w8, w8, #1
stur    w8, [x29, #-16]
b       LBB2_5
LBB2_10:
ldp     x29, x30, [sp, #80]           ; 16-byte Folded Reload
add     sp, sp, #96
ret
.cfi_endproc

                                ; -- End function

.globl _extractDate                  ; -- Begin function extractDate
.p2align    2
_extractDate:                        ; @extractDate
.cfi_startproc
; %bb.0:
sub     sp, sp, #48

```

```

stp    x29, x30, [sp, #32]        ; 16-byte Folded Spill
add    x29, sp, #32
.cfi_def_cfa w29, 16
.cfi_offset w30, -8
.cfi_offset w29, -16
stur   x0, [x29, #-8]
ldur   x0, [x29, #-8]
adrp   x1, l_.str.2@PAGE
add    x1, x1, l_.str.2@PAGEOFF
bl     _strstr
add    x8, x0, #1
str    x8, [sp, #16]
mov    x0, #11
bl     _malloc
str    x0, [sp, #8]
ldr    x0, [sp, #8]
ldr    x1, [sp, #16]
mov    x2, #10
mov    x3, #-1
bl     ____strncpy_chk
ldr    x8, [sp, #8]
strb   wzr, [x8, #10]
ldr    x0, [sp, #8]
ldp    x29, x30, [sp, #32]        ; 16-byte Folded Reload
add    sp, sp, #48
ret
.cfi_endproc

; -- End function

.globl _main                      ; -- Begin function main

```

```

        .p2align    2
_main:                                     ; @main
        .cfi_startproc
; %bb.0:
        stp    x28, x27, [sp, #-32]!      ; 16-byte Folded Spill
        stp    x29, x30, [sp, #16]        ; 16-byte Folded Spill
        add    x29, sp, #16
        sub    sp, sp, #1920
        .cfi_def_cfa w29, 16
        .cfi_offset w30, -8
        .cfi_offset w29, -16
        .cfi_offset w27, -24
        .cfi_offset w28, -32
        adrp   x8, ___stack_chk_guard@GOTPAGE
        ldr    x8, [x8, ___stack_chk_guard@GOTPAGEOFF]
        ldr    x8, [x8]
        stur   x8, [x29, #-24]
        str    wzr, [sp, #52]
        adrp   x0, l_.str.3@PAGE
        add    x0, x0, l_.str.3@PAGEOFF
        adrp   x1, l_.str.4@PAGE
        add    x1, x1, l_.str.4@PAGEOFF
        bl     _fopen
        str    x0, [sp, #40]
        ldr    x8, [sp, #40]
        cbnz   x8, LBB4_2
        b      LBB4_1
LBB4_1:
        adrp   x0, l_.str.5@PAGE

```

```

    add    x0, x0, l_.str.5@PAGEOFF
    bl     _perror
    mov    w8, #1
    str    w8, [sp, #52]
    b      LBB4_6
LBB4_2:
    str    wzr, [sp, #36]
    b      LBB4_3
LBB4_3:                                     ; =>This Inner Loop Header: Depth=1
    ldr    x2, [sp, #40]
    add    x0, sp, #56
    mov    w1, #256
    bl     _fgets
    cbz    x0, LBB4_5
    b      LBB4_4
LBB4_4:                                     ; in Loop: Header=BB4_3 Depth=1
    add    x0, sp, #56
    bl     _extractDate
    str    x0, [sp, #24]
    ldr    w1, [sp, #36]
    ldr    x2, [sp, #24]
    add    x0, sp, #312
    bl     _updateCount
    ldr    x0, [sp, #24]
    bl     _free
    b      LBB4_3
LBB4_5:
    ldr    x0, [sp, #40]
    bl     _fclose

```

```

ldrsw x1, [sp, #36]
add    x0, sp, #312
str    x0, [sp, #16]          ; 8-byte Folded Spill
mov    x2, #16
adrp   x3, _compareEntries@PAGE
add    x3, x3, _compareEntries@PAGEOFF
bl     _qsort
ldr    x0, [sp, #16]          ; 8-byte Folded Reload
ldr    w1, [sp, #36]
bl     _printTopDates
str    wzr, [sp, #52]
b      LBB4_6

```

LBB4_6:

```

ldr    w8, [sp, #52]
str    w8, [sp, #12]          ; 4-byte Folded Spill
adrp   x8, ___stack_chk_guard@GOTPAGE
ldr    x8, [x8, ___stack_chk_guard@GOTPAGEOFF]
ldr    x8, [x8]
ldur   x9, [x29, #-24]
subs   x8, x8, x9
b.ne   LBB4_8
b      LBB4_7

```

LBB4_7:

```

ldr    w0, [sp, #12]          ; 4-byte Folded Reload
add    sp, sp, #1920
ldp    x29, x30, [sp, #16]    ; 16-byte Folded Reload
ldp    x28, x27, [sp], #32    ; 16-byte Folded Reload
ret

```

LBB4_8:

```

bl      ____stack_chk_fail
.cfi_endproc

                ; -- End function

.section      __TEXT,__cstring,cstring_literals

l_.str:                ; @.str
                .asciz "Top 10 Dates:\n"


l_.str.1:              ; @.str.1
                .asciz "%d. %s - %d - %.2lf%%\n"


l_.str.2:              ; @.str.2
                .asciz "["


l_.str.3:              ; @.str.3
                .asciz "01-tiny-access.log"


l_.str.4:              ; @.str.4
                .asciz "r"


l_.str.5:              ; @.str.5
                .asciz "Unable to open file"


.subsections_via_symbols

```

```
1. 09/Oct/2016 - 358 - 10,47%
2. 10/Oct/2016 - 348 - 10,18%
3. 01/Oct/2016 - 347 - 10,15%
4. 20/Oct/2016 - 346 - 10,12%
5. 25/Oct/2016 - 339 - 9,91%
6. 24/Oct/2016 - 337 - 9,85%
7. 12/Oct/2016 - 337 - 9,85%
8. 19/Oct/2016 - 336 - 9,82%
9. 06/Oct/2016 - 336 - 9,82%
10. 02/Oct/2016 - 336 - 9,82%
```

Вивід для 01-tiny-access.log

```
1. 29/Nov/2016 - 35046 - 10,56%
2. 23/Nov/2016 - 34390 - 10,37%
3. 24/Oct/2016 - 33089 - 9,97%
4. 08/Nov/2016 - 32995 - 9,95%
5. 15/Nov/2016 - 32912 - 9,92%
6. 22/Nov/2016 - 32911 - 9,92%
7. 07/Nov/2016 - 32878 - 9,91%
8. 14/Nov/2016 - 32811 - 9,89%
9. 21/Nov/2016 - 32400 - 9,77%
10. 03/Nov/2016 - 32295 - 9,74%
```

Вивід для 05-huge-access.log