CS 221 Advanced Assembly Evaluation Assignment

Lily Larsen

November 19, 2022

1.

a.

https://github.com/Id405/cs-221-eval-assignments/blob/main/eval-6/1a.c

```c
#include <stdint.h>
#include <stdio.h>

// the tail call modified function
uint32_t decaying_sum_t(uint16_t *values, uint16_t length, uint16_t decay,
                        uint32_t value) {
  if (length == 0) {
    return value;
  }
  return decaying_sum_t(values, length - 1, decay,
                        value / decay + values[length - 1]);
}

uint32_t decaying_sum_tail_recursive(uint16_t *values, uint16_t length,
                                     uint16_t decay) {
  return decaying_sum_t(values, length, decay, 0);
}

// the original function for testing purposes
uint32_t decaying_sum(uint16_t *values, uint16_t length, uint16_t decay) {
  if (length <= 0) {
    return 0;
  }
  uint32_t rest = decaying_sum(&values[1], length - 1, decay);
  uint32_t decayed = rest / decay;
  return values[0] + decayed;
}

// test to see if the functions produce the same output on a set of values
int main() {
  uint16_t values[6] = {27, 3, 13, 95, 0, 32};
  uint16_t length = 6;
  uint16_t decay = 5;

  printf("%d\n", decaying_sum(values, length, decay));
  printf("%d\n", decaying_sum_tail_recursive(values, length, decay));
}
```

b.

https://github.com/Id405/cs-221-eval-assignments/blob/main/eval-6/1b.s

```asm
.data
array:
    .half 27
    .half 3
    .half 13
    .half 95
    .half 0
    .half 32
```

```
9
10  .text
11  .globl main
12
13  main:
14      la $a0 array
15          li $a1 6
16          li $a2 5
17          jal decaying_sum
18          li $v0 1
19          syscall
20          li $v0 10
21          syscall
22  decaying_sum:
23          li $a3 0
24  decaying_sum_t:
25      beq $a1 0 return
26      subu $a1 $a1 2
27      divu $a3 $a3 $a2
28      addu $t0 $a0 $a1
29      lhu $t0 ($t0)
30      addu $a3 $a3 $t0
31      j decaying_sum_t
32  return:
33      move $v0 $a3
34      jr $ra
```

2.

https://github.com/Id405/cs-221-eval-assignments/blob/main/eval-6/2.s

```
.data
array:
    .word 27
    .word 3
    .word 13
    .word 95
    .word 0
    .word 32
array2:
    .word 32
    .word 9
    .word 2
    .word 6
    .word 472
    .word 19
array3:
    .align 48
sep: .asciiz ", "

.text
.globl main

main:
    la $a0 array
    la $a1 array2
    la $a2 array3
        li $a3 6
        jal array_product

        li $t0 0
loop_print:
    mul $t1 $t0 8
    addu $t1 $t1 $a2
    lh $a0 4($t1)
    li $v0 1
    syscall
    la $a0 sep
    li $v0 4
    syscall

        addu $t0 $t0 1
    blt $t0 $a3 loop_print

    li $v0 10
    syscall
array_product:
    push $ra
    push $s0
    push $s1
    push $s2
    push $s3
        push $s4
        move $s0 $a0
```

```
54          move $s1 $a1
55          move $s2 $a2
56          move $s3 $a3
57
58          li $s4 0
59  loop:
60          mulu $t0 $s4 4
61          addu $a0 $s0 $t0
62          addu $a1 $s1 $t0
63          mulu $t0 $s4 8
64          addu $t0 $s2 $t2
65
66          jal product
67          sh ($t0) $v1
68          sh 4($t0) $v0
69          addu $s4 $s4 1
70          blt $s4 $s3 loop
71
72          pop $s4
73          pop $s3
74      pop $s2
75      pop $s1
76      pop $s0
77      pop $ra
78      jr $ra
79  product:
80      mul $v0 $a0 $a1
81      mfhi $v1
82      jr $ra
```

3.

https://github.com/Id405/cs-221-eval-assignments/blob/main/eval-6/3.s

```
.data
sunday: .asciiz "sunday"
monday: .asciiz "monday"
tuesday: .asciiz "tuesday"
wednesday: .asciiz "wednesday"
thursday: .asciiz "thursday"
friday: .asciiz "friday"
saturday: .asciiz "saturday"
daytable: .word su mo tu we th fr sa default
string: .align 10

.text
.globl main

main:
    li $a0 2
    la $a1 string
    jal day
    move $a0 $a1
    li $v0 4
    syscall
    li $v0 10
    syscall
day:
    mult $a0 4
    lw $t0 daytable($a0)
    jr $t0
su:
    la $t1 sunday
    j string_copy
mo:
    la $t1 monday
    j string_copy
tu:
    la $t1 tuesday
    j string_copy
we:
    la $t1 wednesday
    j string_copy
th:
    la $t1 thursday
    j string_copy
fr:
    la $t1 friday
    j string_copy
sa:
    la $t1 saturday
    j string_copy
string_copy:
            li $t0 0
string_copy_loop:
    addu $t2 $t1 $t0
    lb $t2 $t2
```

```
54      addu $t3 $a1 $t0
55      sb $t2 $a1
56      bne $t2 0 string_copy_loop
57      li $v0 1
58      jr $ra
59  default:
60      la $v0 0
61      jr $ra
```

4.

https://github.com/Id405/cs-221-eval-assignments/blob/main/eval-6/4.s

```
.data
array:
    .float 10.5
    .float 30.2
           .float 0.9
    .float 0.67

.text
.globl main

main:
    li.s $f0 2.13
          mfc1 $a0 $f0
          la $a1 array
          li $a2 4
        jal accumulator_exponent

accumulator_exponent:
    mfc1 $f0 $a0
        li $t0 0
loop:
    mult $t1 $t0 4
    lh.s $f1 array($t1)
    add.s $f0 $f1
    addu $t0 4
    blt $t0 loop

    mfc1 $v0 $f0
    srl $v0 $v0 23
    and $v0 $v0 255
    jr $ra
```