ICS Homework 6

April 11, 2020

1 Organization

1.1

Consider the following function to copy the contents of one array to another:

```
1 void copy_array(long *src, long *dest, long n) {
2     long i;
3     for (i = 0; i < n; i++)
4         dest[i] = src[i];
5 }
```

Suppose `a` is an array of length 1000 initialized so that each element `a[i]` equals i.

1. What would the array become if call `copy_array(a+1, a, 999)`?
2. What would the array become if call `copy_array(a, a+1, 999)`?
3. Our performance measurements indicate that the call of part a has a CPE of 1.2, while the call of part b has a CPE of 5.0. To what factor do you attribute this performance difference?
4. What performance (CPE) would you expect for the call `copy_array(a, a, 999)`? Please explain your answer.

1.2

The assembly code generated for the compiled loop of combine3 is shown below:

```
1  combine3:
2  # data_t = float, OP = *
3  # i in %rdx, data in %rax, dest in %rbp
4  .L498:                          loop:
5  movss (%rbp), %xmm0             Read from dest
6  mulss (%rax, %rdx, 4), %xmm0    Multiply
7  movss %xmm0, (%rbp)             Store at dest
8  addq $1, %rdx                   Increment i
9  cmpq %rdx, %r12                 Compare i: limit
10 jg .L498                        If >, goto loop
```

Illustrate the code above with data-flow graph like figure 5.14(a) or (b) in CSAPP. You can use "store" to identify operation in line 4.

2 System Software

2.1 Signal

Recall our second problem in exe-5. Please use `sigsuspend` to fix potential bugs in the example code.

2.2 Non-local Jump

Consider the following program:

```
1  #include <setjump.h>
2  sigjmp_buf buf;
3  void handler(int sig) {
4    siglongjmp(buf, 1);
5  }
6  int main() {
7    if (!sigsetjmp(buf, 1)) {
8      Signal(SIGINT, handler);
9      Sio_puts("starting\n");
10   } else
11     Sio_puts("restarting\n");
12
13   while(1) {
14     Sleep(1); Sio_puts("processing...\n");
15   }
16   exit(0); /* Control never reaches here */
17 }
```

Please give an example output of the program when we launch it and press `Ctrl+C` several times.