CSC 209 Review 9 Solution

1. a) 0

   b)

   Notes

   - a) is 0 because (i >> 1 + j >> 1 = i >> 10 >> 1 = 0)
   - **Bitwise Shift Operators**
     - has lower precedence than arithematic operators

       Example:

       i << 2 + 1 means i << (2+1) and not (i << 2) + 1

     - << : Left Shift
     - >> : Right Shift
     - *Tip:* Always shift only on <u>unsigned</u> numbers for portability

       Example

```
unsigned short i, j;

i = 13;       /* i is now 13 (binary 0000000000001101) */   Shifts to left
j = i << 2;   /* j is now 52 (binary 0000000000110100) */
j = i >> 2;   /* j is now  3 (binary 0000000000000011) */
                               Shifts to right

As these examples show, neither operator modifies its operands. To modify a vari-
able by shifting its bits, we'd use the compound assignment operators <<= and
>>=:

i = 13;       /* i is now 13 (binary 0000000000001101) */
i <<= 2;      /* i is now 52 (binary 0000000000110100) */
i >>= 2;      /* i is now 13 (binary 0000000000001101) */
```

     - >>= / <<= : Are bitwise shift equivalent of + =