

# CSC 209 Review 5 Solution

August 21, 2020

1. a) 14  
b) 34  
c) 4  
d) true  
e) false

## Notes

- **Pointer Arithmetic**

- Adding an integer to a pointer

## Example



- Subtracting an integer from a pointer

### Example



- Subtracting one pointer from another

### Example



### • Comparing pointers

- Can compare pointers using relational operators (i.e. `<`, `<=`, `>`, `>=`) and the equality operators (i.e. `==`, `!=`)
- Returns 1 if **true** and 0 if **false**

### Example

```
p = &a[5];
q = &a[1];
```

`p <= q` is 0 and `p >= q` is 1

2. `low` and `high` are memory addresses.

So, `low + high` is out of bound, and it could potentially point to an undesirable or wrong value.

To fix this, we subtract the from high value to the low value:

$$\text{middle} = \frac{\text{low} + \text{high}}{2} \quad (1)$$

### 3. Notes

- **Combining the \* and ++ Operators**

- `*p++` or `*p++` → Value of expression is `*p` before increment; increment `p` later
- `(*p)++` → Value of expression is `*p` before increment; increment `*p` later
- `+++p` or `*(++p)` → Increment `p` first; value of expression is `*p` after increment
- `++*p` or `++(*p)` → Increment `*p` first; value of expression is `*p` after increment

#### Example

`a[i++] = j`

Means assign the value `j` to `a[i]` before increment