

Fall 2022 Exam 1

Q1:

a) $O(n^2)$ because for the first while loop is $O(n)$ and for the second while loop is $O(n)$, but since the second while loop is nested under the first one you multiply the two $O(n)$ s together to make n^2

Q2:

b) $O(N \log N)$ because the first while loop is just $O(n)$, but the second loop is $O(\log N)$ because of the $k *= 3$ line which makes the run time shorter. The two multiplied together make $O(N \log N)$

Q3:

c) $O(n^2)$ because the first for loop runs $O(n)$ times and the second for loop runs $O(n)$. Since they are nested together that makes it $O(n^2)$

5	6	7	8	9	10	11	12	13	14
32	64	128	256	512	1024	2048	4096	8192	16384

Q4: $\log_2(1000) \sim 10$ $2^{10} = 1000$

e) $O(N \log_2(N))$

$$2000 \log_2(2000) = 10$$

$$16000 \approx 101 \text{ seconds}$$

$$\frac{2000 \log_2 2000}{16000 \log_2 16000} = \frac{10}{x}$$

$$2000 \log_2 2000 \cdot x = 160000 \log_2 16000$$

$$\log_2 2000 \cdot x = 80 \log_2 16000$$

$$11x = 80(14)$$

$$x = \frac{1120}{11}$$

$$\begin{array}{r} 80 \\ 14 \\ \hline 320 \\ 80 \\ \hline 1120 \end{array} \quad \begin{array}{r} 101 \\ 11 \overline{) 1120} \\ \hline \end{array}$$

Q5:

c) `int n;`

`float floats[] = {1.2, 3.4, 9.9};` prints: 2 4 10

`for (int i = 0; i < 3; i++) {`

`n = (int) (floats[i] * 0.5);`

`printf("%d", n);`

3

$$1.2 \times 0.5 = 0.6$$

$$3.4 \times 0.5 = 1.7$$

$$9.9 \times 0.5 = 4.95$$

Q6:

d)

i) 312

ii) false

iii) 0x1001

iv) 0x1001

v) H

vi) a causes a memory access error because the word

only has 6 letters, but in the print statement it's

trying to access 7 which is outside the allocated

memory space. ↑ the letter in the 7th position in the array

Q7)

```
char* utstructstr(UTString* str1, UTString* str2){
    if(str2.length == 0){
        return str1;
    }
    int temp = 0;
    if(str1.length == 0 && str2.length == 0){
        return;
    }
    for(int x=0; x<str1.length; x++){
        char* result; int count = x; int check = 0;
        for(int y=0; y<str2.length; y++){
            if(str1.string[count] == str2.string[y]){
                count++; check++;
            }
            if(check == str2.length){
                return str1+x;
            }
        }
    }
    return NULL;
}
```

Q8)

a) 16 bytes

```
Q9) Ccard* bigBalances(Ccard* ccards, uint32_t n){
    Ccard* big = (Ccard*) malloc(sizeof(Ccard)*n);
    int count = 0;
    for(int x=0; x<n; x++){
        if(ccards[x].balance > 1000){
            big[count].customer_name = ccards[x].customer_name;
            big[count].balance = ccards[x].balance;
            count++;
        }
    }
    return big;
}
```

Q10:

```
void free_cards(Ccard* ccards, uint32_t n){
    for(int x=0; x<n; x++){
        free(ccards[x]);
    }
    free(ccards);
}
```