University of Nevada, Las Vegas Computer Science 477/677 Fall 2022

Assignment 1: Due Friday September 9 2022 11:59 PM

Name:_____

You are permitted to work in groups, get help from others, read books, and use the internet. Your answers must be written in a pdf file and uploaded to canvas, by midnight September 9th. Your file must not be unnecessarily long. If you have any questions, or you are having trouble uploading the assignment you may email the grader, Shizhao Wang, at `wangs12@unlv.nevada.edu`. You may also send me email to ask questions.

1. Problem 0.1 on page 8 of the textbook. In each of the following situations, write O, Ω. Θ in the blank.

   (a) $n - 100 =$ _____ $(n - 200)$

   (b) $n^{1/2} =$ _____ $(n^{2/3})$

   (c) $100n + \log n =$ _____ $(n + \log^2 n)$

   (d) $n \log n =$ _____ $(10n + \log(10n))$

   (e) $\log(2n) =$ _____ $(\log(3n))$

   (f) $10 \log n =$ _____ $(\log(n^2))$

   (g) $n^{1.01} =$ _____ $(n \log^2 n)$

   (h) $n^2 / \log n =$ _____ $(n \log^2 n)$

   (i) $n^{0.1} =$ _____ $(\log^2 n)$

   (j) $(logn)^{\log n} =$ _____ $(n / \log n)$

   (k) $\sqrt{n} =$ _____ $(\log^3 n)$

   (l) $n^{1/2} =$ _____ $(5^{log_2 n})$

   (m) $n2^n =$ _____ (3^n)

   (n) $2^n =$ _____ (2^{n+1})

   (o) $n! =$ _____ (2^n)

   (p) $\log n^{\log n} =$ _____ $(2^{(\log_2 n)^2})$

(q) $\sum_{i=1}^{n} i^k =$ ------(n^{k+1})

2. Work problem 0.3(c) on page 9 of the textbook.[1]

   The Fibonacci numbers $F_0, F_1, F_2, \ldots$ are defined by the rules
   $F_0 = 0$
   $F_1 = 1$
   $F_n = F_{n-2} + F_{n-1}$ for $n \geq 2$

   What is the largest constant C for which $F_n = \Omega(C^n)$? Hint: C is a solution of a quadratic equation.

3. Consider the following C++ program.

```
void process(int n)
 {
  cout << n << endl;
  if(n > 1) process(n/2);
  cout << n%2;
 }

int main()
 {
  int n;
  cout << "Enter a positive integer: ";
  cin >> n;
  assert(n > 0);
  process(n);
  cout << endl;
  return 1;
 }
```

   The last line of the output of `process(n)` is a string of bits. What does this bitstring represent?

4. The recursive algorithm implemented below as a C++ function is used as a subroutine during the calculation of the level payment of an amortized loan. What does it compute?

```
float squre(float x)
 {
  return x*x;
 }

float mystery(float x, int k)
 {
  if (k == 0) return 1.0;
  else if(x == 0.0) return 0.0;
```

[1] I have rewritten the problem slightly.

```
  else if (k < 0) return 1/mystery(x,-k);
  else if (k%2) return x*mystery(x,k-1);
  else return mystery(squre(x),k/2);
}
```

5. How many times will each of these code fragments print "Hello, world!" Use Θ notation.

(a)
```
for (int i=1; i < n; i++)
    for (int j=i; j > 0; j--)
      cout << "Hello, world!" << endl;
```

(b)
```
for (int i=1; i < n; i=2*i)
    for (int j=i; j < n; j++)
      cout << "Hello, world!" << endl;
```

(c)
```
for (int i=1; i < n; i = 2*i)
    for (int j=1; j < i; j++)
      cout << "Hello, world!" << endl;
```

(d)
```
for (int i=1; i < n; i++)
    for (int j=1; j < i; j = j*2)
      cout << "Hello, world!" << endl;
```

(e)
```
for (int i=1; i < n; i++)
    for (int j=i; j < n; j = j*2)
      cout << "Hello, world!" << endl;
```

(f)
```
for (int i=2; i < n; i = i*i)
    cout << "Hello, world!" << endl;
```

(g)
```
for (int i=1; i*i < n; i++)
    cout << "Hello, world!" << endl;
```

(h)
```
for (int i=n; i > 1; i = i/2)
    for (int j=1; j < i; j=2*j)
      cout << "Hello, world!" << endl;
```