Guideline

Due Date: Thursday, 2023-11-09, by 23:59.

Upload your answers as a singular PDF to Brightspace.

If you're writing by hand, please ensure your handwriting is legible.

Multiple submissions are possible before the due time; the last submission will be graded.

You may or may not follow the template below when writing your proof using mathematical induction. It is provided for your convenience.

```
    Proof.
      Let P(n) denote ______. We want to prove P(n) is true for ______.
      We use mathematical induction to proceed.
      Base Step: We want to prove _______.
         ________ (put your proof of the above goal in the base step here)

      Inductive Step: We want to prove ______.
        ________ (put your proof of the above goal in the inductive step
here)
    QED.
```

Exercise 1 (10 points)

Use mathematical induction to prove that $\Sigma_{i=1 \text{ to } n}\ i^3 = n^2(n+1)^2/4$ for every positive integer n >= 1.

Exercise 2 (20 points)

Use mathematical induction to prove that $\Sigma_{i=1 \text{ to } n}\ (8i-5) = 4n^2 - n$ for every positive integer n >= 1.

Exercise 3 (20 points)

The triangle inequality states that for all real numbers a and b, |a + b| <= |a| + |b|. Use the triangle inequality and mathematical induction to prove:

For any n real numbers a_1, a_2, ..., and a_n,

$$|\Sigma_{i=1}^{n} a_i| \leq \Sigma_{i=1}^{n} |a_i|$$

Exercise 4 (20 points)

Let f be a sequence defined recursively as follows:

- $f_1 = 1$, and
- $f_k = f_{k-1} + 2^k$ for all integers k >= 2

1. Write out f_k for k = 1, 2,..., 5.
2. Follow the pattern in #1 to guess an explicit form of the sequence. (Remind: An explicit form is a formula that looks like $a_n=n(n+1)/2$, which should involve no recursions.)
3. Prove that your explicit form corresponds to the original recursive definition. (Remind: You can plug the explicit form into the recursive definition. Mathematical induction would be unnecessary for this exercise.)

Exercise 5 (20 points)

A specific computer program executes twice as many operations when it runs with an input of size k as when it runs with an input of size k – 1 (where k is an integer and k > 1). When the program runs with an input of size 1, it executes seven operations. Let a_n be the number of operations when it runs with an input of size n. Please (1) first figure out a recursive form of a_n, together with a base case. (2) Get the explicit form. (3) Confirm the explicit one is correct with regard to the recursive form, and (4) calculate a_{25}.