- **Groups of up to three people can submit joint solutions.** Each problem should be submitted by exactly one person, and the beginning of the homework should clearly state the Gradescope names and email addresses of each group member. In addition, whoever submits the homework must tell Gradescope who their other group members are.
- **Submit your solutions electronically on the course Gradescope site as PDF files.** Submit a separate PDF file for each numbered problem. If you plan to typeset your solutions, please use the LaTeX solution template on the course web site. If you must submit scanned handwritten solutions, please use a black pen on blank white paper and a high-quality scanner app (or an actual scanner, not just a phone camera).
- You are *not* required to sign up on Gradescope (or Piazza) with your real name and your illinois.edu email address; you may use any email address and alias of your choice. However, to give you credit for the homework, we need to know who Gradescope thinks you are. **Please fill out the web form linked from the course web page.**

Some important course policies

- **You may use any source at your disposal** – paper, electronic, or human-but you *must* cite *every* source that you use, and you *must* write everything yourself in your own words. See the academic integrity policies on the course web site for more details.
- For any question, you can write **IDK** ("I Don't Know") and get 25% of the points for the question. You would get the points if and only if IDK is the only content of your answer. Any answer containing "IDK" anywhere and **any** additional text would immediately get a zero.
- **Avoid the Three Deadly Sins!** Any homework or exam solution that breaks any of the following rules will be given an *automatic zero*, unless the solution is otherwise perfect. Yes, we really mean it. We are not trying to be scary or petty (Honest!), but we do want to break a few common bad habits that seriously impede mastery of the course material.
  - Always give complete solutions, not just examples.
  - Always declare all your variables, in English. In particular, always describe the specific problem your algorithm is supposed to solve.
  - Never use weak induction.
  - Always give credit to outside sources! (Yes, we are not good in counting.)

See the course web site for mo information.

If you have any questions about these policies,
please do not hesitate to ask in class, in office hours, or on Piazza.

---

1 (100 PTS.) Repeat that.

1.A. Let $x_1, \dots, x_n$ be a sequence of integer numbers, such that $\alpha \leq x_i \leq \beta$, for all i, where α, β are some integer numbers. Prove that there are at least $\lceil n/(\beta - \alpha + 1) \rceil$ numbers in this sequence that are all equal.

1.B. Let $\mathsf{G} = (\mathsf{V}, \mathsf{E})$ be an undirected graph. Unless we say otherwise, a graph has no loops or parallel edges. Prove, using (A), that if $|\mathsf{V}| \geq 2$ there are two distinct nodes u and v such that degree of u is equal to degree of v. Recall that the degree of a node x is the number of edges incident to x.

1.C. Prove that if all vertices in G are of degree at least one, then there is a (simple) path between two distinct nodes u and v such that degree of u is equal to degree of v.

2 (100 PTS.) Mix this.

The *sort*, w^s, of a string $w \in \{0,1\}^*$ is obtained from w by sorting its characters. For example, $010101^s = 000111$. The sort function is formally defined as follows:

$$w^s := \begin{cases} \epsilon & \text{if } w = \epsilon \\ 0x^s & \text{if } w = 0x \\ x^s 1 & \text{if } w = 1x \end{cases}$$

The ***merge*** function, evaluated in order from top to bottom, is

$$m(x,y) := \begin{cases} y & \text{if } x = \varepsilon \\ x & \text{if } y = \varepsilon \\ 0m(x',y) & \text{if } x = 0x' \\ 0m(x,y') & \text{if } y = 0y' \\ 1m(x',y) & \text{if } x = 1x' \end{cases}$$

For example, we have $m(10,10) = 1010$, $m(10,010) = 01010$, and $m(010,0001100) = 0000101100$.

For a string $x \in \{0,1\}^*$, let $\#_0(x)$ and $\#_1(y)$ be the number of 0s and 1s in x, respectively. For example, $\#_0(0101010) = 4$ and $\#_1(0101010) = 3$.

2.A. (**Not for submission.**) Prove by induction that for any string $w \in \{0,1\}^*$ we have that $w^s \in 0^*1^*$.

2.B. Prove by induction that for any string $w \in \{0,1\}^*$ we have that $\#_0(w) = \#_0(w^s)$. Conclude that $\#_1(w) = \#_1(w^s)$ and $|w| = \left|w^s\right|$, for any string w.

2.C. Prove by induction that for any two strings $x, y \in \{0,1\}^*$ we have that

$$\#_0(m(x,y)) = \#_0(x) + \#_0(y).$$

(**Hint:** Do induction on $|x| + |y|$.)
Conclude that $\#_1(m(x,y)) = \#_1(x) + \#_1(y)$. and $\left|m(x,y)\right| = \left|x\right| + \left|y\right|$.
[This part is somewhat tedious if you write carefully all the details out explicitly. Avoid repetition by stating that you are (essentially) repeating an argument that was already seen in the proof.]

2.D. Prove by induction that for any two strings x, y of the form 0^*1^*, we have that $m(x, y)$ is of the form 0^*1^*.

2.E. Prove (using the above) that $(x \bullet y)^s = m(x^s, y^s)$ for all strings $x, y \in \{0, 1\}^*$.

3 (100 PTS.) Walk on the grid.

Let $p_0 = (x_0, y_0)$ be a point on the positive integer grid (i.e., x_0, y_0 are positive integer numbers). A point (x, y) is ***good*** if $x = y$ or $x = 0$ or $y = 0$. For a point $p = (x, y)$ its ***successor*** is defined to be

$$\alpha(p) = \begin{cases} (x, y - x - 1) & y > x \qquad \text{(vertical move)} \\ (x - y - 1, y) & x > y \qquad \text{(horizontal move).} \end{cases}$$

Consider the following sequence $W(p_0) = p_0, p_1, \ldots$ computed for p_0. In the ith stage of computing the sequence, if p_{i-1} is good then the sequence is done as we arrived to a good location. Otherwise, we set $p_i = \alpha(p_{i-1})$.

3.A. Prove, by induction, that starting with any point p on the positive integer grid, the sequence $W(p)$ is finite (i.e., the algorithm performs a finite number of steps before stopping).

3.B. (Harder.) Given such a sequence, every step between two consecutive points is either a vertical or a horizontal move. A ***run*** is a maximal sequence of steps in the walk that are the same (all vertical or all horizontal). Prove that starting with a point $p = (x, y)$ there are at most $O(\log x + \log y)$ runs in the sequence $W(p)$.