

# CS 70 Midterm 1 Cheat Sheet

Suraj Rampure, CSM Junior Mentor

The following material comes from the official CS 70 notes and CSM worksheets, along with my own personal intuition.

## Graphs

A **path** is a sequence of edges.

A **simple path** is a sequence of edges with no repeated vertices.

A **walk** is a sequence of edges that may have repeated vertices/edges.

A **tour** is a walk that starts and ends at the same vertex.

A **cycle** is a tour that does not repeat vertices other than the start/end vertex.

A **tournament** is a complete graph but with directed edges, i.e. for all pairs of vertices in the graph  $u, v$  there exists the edge  $u \rightarrow v$  or  $v \rightarrow u$ .

A **Hamiltonian path** is a path that visits every vertex exactly once.

A **Hamiltonian tour/cycle** is a cycle that goes through every vertex exactly once.

A **Eulerian path** is a path that traverses every edge exactly once.

A **Eulerian tour** is a Eulerian path that starts and ends on the same edge.

**Euler's theorem:** A Eulerian tour exists in every undirected connected graph where every vertex has an even degree.

A corollary – Eulerian paths exist when at most two vertices have odd degree.

## Hypercubes

- $2^n$  vertices,  $n \cdot 2^{n-1}$  edges
- $n$ th degree built by making two copies of  $n - 1$ th degree hypercube and creating an edge between corresponding elements
- Numbering with bitstrings – vertices connected by an edge are different in only one bit position
- Two-colorable

## Stable Marriage

- Traditional Marriage Algorithm (TMA) – men propose, women reject

- TMA is Male optimal, female pessimal
- Algorithm terminates in a stable pairing
- Improvement Lemma – the fortunes of the women in the TMA increase each day (i.e. a woman likes her partner-on-a-string on day  $k + 1$  at least as much as she liked her partner-on-a-string on day  $k$ ; vice versa for men)

## Bijections

A function is **injective** (one-to-one) if, for any elements of the domain  $a, b \in D$ ,  $f(a) = f(b) \implies a = b$ . In other words, a function is injective if there does not exist an element in the range/codomain that is pointed to by two different elements of the range.

$f(x) = x^3$  is injective, while  $f(x) = x^2$  is **not** injective; for example,  $f(3) = f(-3)$ , but  $3 \neq -3$ .

A function is **surjective** (onto) if, for all elements of the range  $y \in R$ ,  $\exists x \mid y = f(x)$ . In other words, a function is surjective if element in the range is mapped to by some element in the domain.

A function is then **bijective** if it is both injective and surjective.

## Modular Arithmetic

When dealing with operations **mod**  $p$ , we mean "taking the remainder when divided by  $p$ ". For example,  $13 + 25 \pmod{3} \equiv 38 \pmod{3} \equiv 36 + 2 \pmod{3} \equiv 2 \pmod{3}$ . Note that we could've obtained the same result by taking **mod** 3 in the beginning –  $13 \equiv 1 \pmod{3}$ ,  $25 \equiv 1 \pmod{3}$ ,  $1 + 1 \equiv 2 \pmod{3} \implies$  the order in which we take our **mod** does not matter.

$\mathbb{Z}_p$  denotes that we are working in the field of integers **mod**  $p$ , meaning that the only integers that exist in the set  $\mathbb{Z}_p$  are  $0, 1, \dots, p - 1$ . When dealing with  $\mathbb{Z}_p$ , we must take all operations **mod**  $p$ .

**Multiplicative Inverses** – The multiplicative inverse of some integer  $n$  in **mod**  $p$  is defined as follows:  $x$  is the multiplicative inverse of  $n$  in **mod**  $p$  iff

$$nx \equiv 1 \pmod{p}$$

Furthermore, the multiplicative inverse exists iff

$$\gcd(n, p) = 1$$

The problem of finding a multiplicative inverse can be written as finding an integral solution to the equation  $nx + py = 1$ .

For determining these inverses, on the midterm, the almighty guess and check should suffice, but just in case:

```

1 algorithm extended-gcd (x, y)
2   if y = 0 then return (x, 1, 0)
3   else
4     (d, a, b) := extended-gcd(y, x mod y)
```

```
5      ...      return ((d, b, a - div(x,y) * b))
```

Where in the above,  $\text{div}(x, y)$  represents the floor of the division  $x/y$ .

## Fermat's Little Theorem

FLT states, for any prime  $p$  and integer  $a < p$ , the following holds:

$$a^{p-1} \equiv 1 \pmod{p}$$

This makes several calculations easier:

$$7^{1000} \pmod{11} \equiv (7^{10})^{100} \pmod{11} \equiv (7^{10} \pmod{11})^{100} \pmod{11} \equiv 1^{100} \pmod{11} \equiv 1 \pmod{11}$$

Another trick to making arithmetic calculations easier – check if the exponential base is one greater or one less than the modulo base. For example:

$$18^{4096} \pmod{17} \equiv 1^{4096} \pmod{17} \equiv 1 \pmod{17}$$

## Chinese Remainder Theorem

Given

$$x \equiv a_1 \pmod{n_1}$$

$$x \equiv a_2 \pmod{n_2}$$

$$\vdots$$

$$x \equiv a_k \pmod{n_k}$$

The solution to  $x$  is

$$x = \left( \sum_{i=1}^k a_i b_i \right) \pmod{N}$$

where  $N = n_1 n_2 \dots n_k$ , and  $b_i$  is defined as  $b_i = \frac{N}{n_i} \text{inv}(N/n_i, n_i)$ , where  $\text{inv}(N/n_i, n_i)$  is the multiplicative inverse of  $\frac{N}{n_i}$  taken in modulo  $n_i$ . A sample application of this is as follows:

Consider  $x \equiv 3 \pmod{4}$  and  $x \equiv 5 \pmod{13}$ . Then,

$$N = 4 * 13 = 52$$

$$b_1 = 13 * \text{inv}(13, 4) = 13 * 3 = 39$$

$$b_2 = 4 * \text{inv}(4, 13) = 4 * 10 = 40$$

Our solution for  $x$  is then

$$\begin{aligned}x &= 3 * 26 + 5 * 40 \pmod{52} = 52 + 26 + 200 \pmod{52} \\&= 226 \pmod{52} = 18 \pmod{52}\end{aligned}$$