Objectives

Review some structured graphs and study their properties.

- Complete graphs
- Cycles
- Meshes and torus
- Hypercubes
- Trees
Complete graphs (or cliques)

Definition.
Each vertex of $K_n$ is connected to all the other vertices.

- Connected ($D = 1$)
- Regular graph ($\delta = n - 1$)
- Number of edges $\sum_{1 \leq k \leq n-1} k = \frac{(n)(n-1)}{2}$
Rings or Cycles

Definition.
Each vertex of $C_n$ has exactly one predecessor and one successor.

Coding of edges

\[ \{\{i, \ i + 1 \mod n\} \mid i \in \{0, 1, \ldots, n-1\}\}. \]

- Connected ($D = \lfloor \frac{n}{2} \rfloor$)
- Regular graph ($\delta = 2$)
- Number of edges $n$
An interesting observation

**Proposition.**
If every vertex of a graph $G$ has degree $\geq 2$, then $G$ contains a cycle.
An interesting observation

Proposition.
If every vertex of a graph $G$ has degree $\geq 2$, then $G$ contains a cycle.

Proof
Let us assume by contradiction that we have a cycle-free graph $G$ all of whose vertices have degree $\geq 2$.

Let us view graph $G$ as a park where every vertex of $G$ is a statue, and every edge is a path between two statues. The fact that every vertex of $G$ has degree $\geq 2$ means that if we take a stroll through $G$, then every time we leave a vertex $v \in V$, we can use a different edge/path than we used when we came to $v$. 
Meshes and Torus

Definition.
Cartesian product of paths/cycles.

Coding of meshes (vertices and edges).

\[
\{1, 2, \ldots, m\} \times \{1, 2, \ldots, n\} \\
\{\langle i, j \rangle \mid [i \in \{1, 2, \ldots, m\}], [j \in \{1, 2, \ldots, n\}]\}
\]

Torus is obtained by adding the wraparound links...
Example for $n = 4$
Properties of the square torus with \( n \) vertices

\[ \sqrt{n} \text{ by } \sqrt{n} \]

- Connected (diameter \( D = \Theta(\sqrt{n}) = 2 \cdot \lceil \frac{\sqrt{n}}{2} \rceil \))
- Regular graph (degree \( \delta = 4 = \Theta(1) \))
- Number of edges \( 2n = \Theta(n) \)
Hypercubes

Motivation:
build a graph with a trade-off between the degree and the diameter.
Motivation: build a graph with a trade-off between the degree and the diameter.

Recursive Definition.

- The order-0 boolean hypercube, $H_0$, has a single vertex, and no edges.
- The order-($k + 1$) boolean hypercube, $H_{k+1}$, is obtained by taking two copies of $H_k$ ($H_k^{(1)}$ and $H_k^{(2)}$), and creating an edge that connects each vertex of $H_k^{(1)}$ with the corresponding vertex of $H_k^{(2)}$. 

Construction
The next dimension

Representation of $H_4$
Coding

A natural binary coding
The coding from the vertices is naturally in the binary system.

The coding of two adjacent vertices is obtained by flipping only one bit.
Characteristics of Hypercubes

The number of vertices is a power of 2: \( n = 2^k \) \((k = \log_2(n))\)

- Diameter \( D_n = k \)
- Degree \( \delta_n = k \)
- Number of edges?
Characteristics of Hypercubes

The number of vertices is a power of 2: \( n = 2^k \ (k = \log_2(n)) \)

- Diameter \( D_n = k \)
- Degree \( \delta_n = k \)
- Number of edges?

\( H_{k+1} \) is obtained by two copies of \( H_k \) plus \( 2^k \) edges for linking each relative vertex, thus:

\[ N_{k+1} = 2 \times N_k + 2^k \] starting at \( N_0 = 0 \)

\[ N_k = k \times 2^{k-1} \]
Graph isomorphism

The difficulty here is that there are many ways to draw a graph...

Property.
The hypercube $H_4$ is identical to the 4 by 4 torus.
Graph isomorphism

The difficulty here is that there are many ways to draw a graph...

Property.
The hypercube $H_4$ is identical to the 4 by 4 torus.
The proof is by an adequate coding of the vertices/edges.
Coding schemes

The following figure (left) depicts this coding of a vertex and its neighbors.
The (almost) full picture
Gray Codes

Cultural aside
Let us present the most popular code, namely, the **Reflected Gray code**

The 1-bit Gray code is simply 0 and 1. The next one (for 2-bits) is obtained by mirroring the 1-bit code and prefix it by 0 and 1. The next ones are obtained similarly.
Gray Code

How to obtain the Gray code from the binary code?
Gray code can easily be determined from the classical binary representation as follows:

\[(x_{n-1}x_{n-2}...x_1x_0)_2\]

shift right:

\[(0x_{n-1}x_{n-2}...x_1)_2\]

Take the exclusive OR (bit-to-bit) between the binary code and its shifted number:

\[(x_{n-1}(x_{n-2} \oplus x_{n-1})...(x_0 \oplus x_1))_G\]

For instance the binary code of 5 = (00101)_2 is

\[(0 \oplus 0)(0 \oplus 0)(0 \oplus 1)(1 \oplus 0)(0 \oplus 1) = (00111)_G.\]
Algebraic properties

```plaintext
00001  00001
00010  00011
00011  00010
00100  00110
```

```
00101  00111
00110  00101
00111  00100
01000  01100
01001  01101
01010  01111
01011  01110
01100  01010
01101  01011
```

...  ...
```