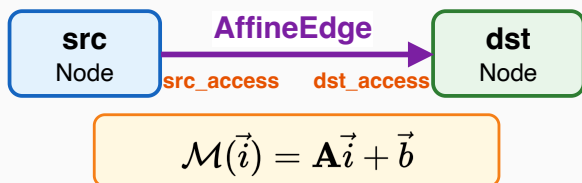## (a) AffineEdge

```
class AffineEdge {
  Node src, dst;
  AffineMap src_access;
  AffineMap dst_access;
};
```

```
class AffineMap {
  Matrix A;
  Vector b;
  vector<IVar> ivars;
};
```

src Node → **AffineEdge** → **dst** Node
src_access   dst_access

$$\mathcal{M}(\vec{i}) = \mathbf{A}\vec{i} + \vec{b}$$

(b) AffineNode

```
union AffineNode {
  Buffer buf;
  Task task;
  AffineGraph subgraph;
  vector<AffineEdge> in_edges();
  vector<AffineEdge> out_edges();
};
```

Buffer GMEM / SMEM / REG   **Task** MMA, Softmax...   **AffineGraph** (nested)

(c) AffineGraph

```
class AffineGraph {
  vector<Node> nodes;
  vector<AffineEdge> edges;
  vector<AffineEdge>
in_edges;
  vector<AffineEdge>
out_edges;
  void topo_sort();
  void emit_task();
};
```

AffineGraph

in_edges

A GMEM
B GMEM

nodes

Tiles SMEM → **MMA** Task → **C** GMEM

edges

out_edges