

```
def simplify(*args):  
    return ''.join(  
        map(lambda x: '0' if isinstance(x, Var) else '1',  
            args))
```

```
lookups = {  
    '000': lambda g, e, a, v: all(g.eav),  
    '001': lambda g, e, a, v: double(g.vea, v),  
    '010': lambda g, e, a, v: double(g.ave, a),  
    '011': lambda g, e, a, v: single(g.ave, a, v),  
    '100': lambda g, e, a, v: double(g.eav, e),  
    '101': lambda g, e, a, v: single(g.vea, v, e),  
    '110': lambda g, e, a, v: single(g.eav, e, a),  
    '111': lambda g, e, a, v: exists(g.eav, e, a, v)}
```

```
def merge(a, b): return a + b
```

```
def all(idx):  
    result = []  
    for a, bm in idx.items():  
        for b, cs in bm.items():  
            for c in cs:  
                result.append([a, b, c])  
    return result
```

```
def double(idx, a):  
    bm = idx.get(a)  
    if bm == None: return []  
    result = []  
    for b, cs in bm.items():  
        for c in cs:  
            result.append([b, c])  
    return result
```

```
def single(idx, a, b):  
    bm = idx.get(a)  
    if bm == None: return []  
    cs = bm.get(b)  
    if cs == None: return []  
    return [[ck] for ck in cs]
```

```
def exists(idx, a, b, c):  
    if None != idx.get(a) and None != idx[a].get(b) and c in idx[a][b]:  
        return [[]]  
    else:  
        return []
```