```
UpdateGraph(Batch);
for(edge ∈ Batch){
 if(edge maps (u0, u1)) M1(u0, u1)
 if(edge maps (u0, u2)) M2(u0, u2)
 if(edge maps (u0, u3)) M3(u0, u3)
 if(edge maps (u1, u3)) M4(u1, u3)
 if(edge maps (u2, u3)) M5(u2, u3)}
```

```
//ΔM1=ΔR1⋈R'2⋈R'3⋈R'4⋈R'5
void ΔM1(x0, x1){
 // ⋈ R'2 (vertex u2)
 for (x2 ∈ N'(x0)){
  // ⋈ R'3 ⋈ R'4 ⋈ R'5 (vertex u3)
  for (x3 ∈ N'(x0) ∩ N'(x1) ∩ N'(x2)){
   output(x0, x1, x2, x3); } } }
```

```
//ΔM2=R1⋈ΔR2⋈R'3⋈R'4⋈R'5
void ΔM2(x0, x2){
 // ⋈ R1 (vertex u1)
 for (x1 ∈ N(x0)){
  // ⋈ R'3 ⋈ R'4 ⋈ R'5 (vertex u3)
  for (x3 ∈ N'(x0) ∩ N'(x1) ∩ N'(x2)){
   output(x0, x1, x2, x3); } } }
```

```
//ΔM3=R1⋈R2⋈ΔR3⋈R'4⋈R'5
void ΔM3(x0, x3){
//⋈ R1 ⋈ R'4 (vertex u1)
 for (x1 ∈ N(x0) ∩ N'(x3)){
  //⋈ R2 ⋈ R'5 (vertex u2)
  for (x2 ∈ N(x0) ∩ N'(x3)){
   output(x0, x1, x2, x3); } } }
```

```
//ΔM4=R1⋈R2⋈R3⋈ΔR4⋈R'5
void ΔM4(x1, x3){
 // ⋈ R1 ⋈ R3 (vertex u0)
 for (x0 ∈ N(x1) ∩ N(x3) ){
  // ⋈ R2 ⋈ R'5 (vertex u2)
  for (x2 ∈ N(x0) ∩ N'(x3)){
   output(x0, x1, x2, x3); } } }
```

```
//ΔM5=R1⋈R2⋈R3⋈R4⋈ΔR5
void ΔM5(x2, x3){
 // ⋈ R2 ⋈ R3 (vertex u0)
 for (x0 ∈ N(x2) ∩ N(x3)){
  // ⋈ R1 ⋈ R4 (vertex u1)
  for (x1 ∈ N(x0) ∩ N(x3)){
   output(x0, x1, x2, x3); } } }
```