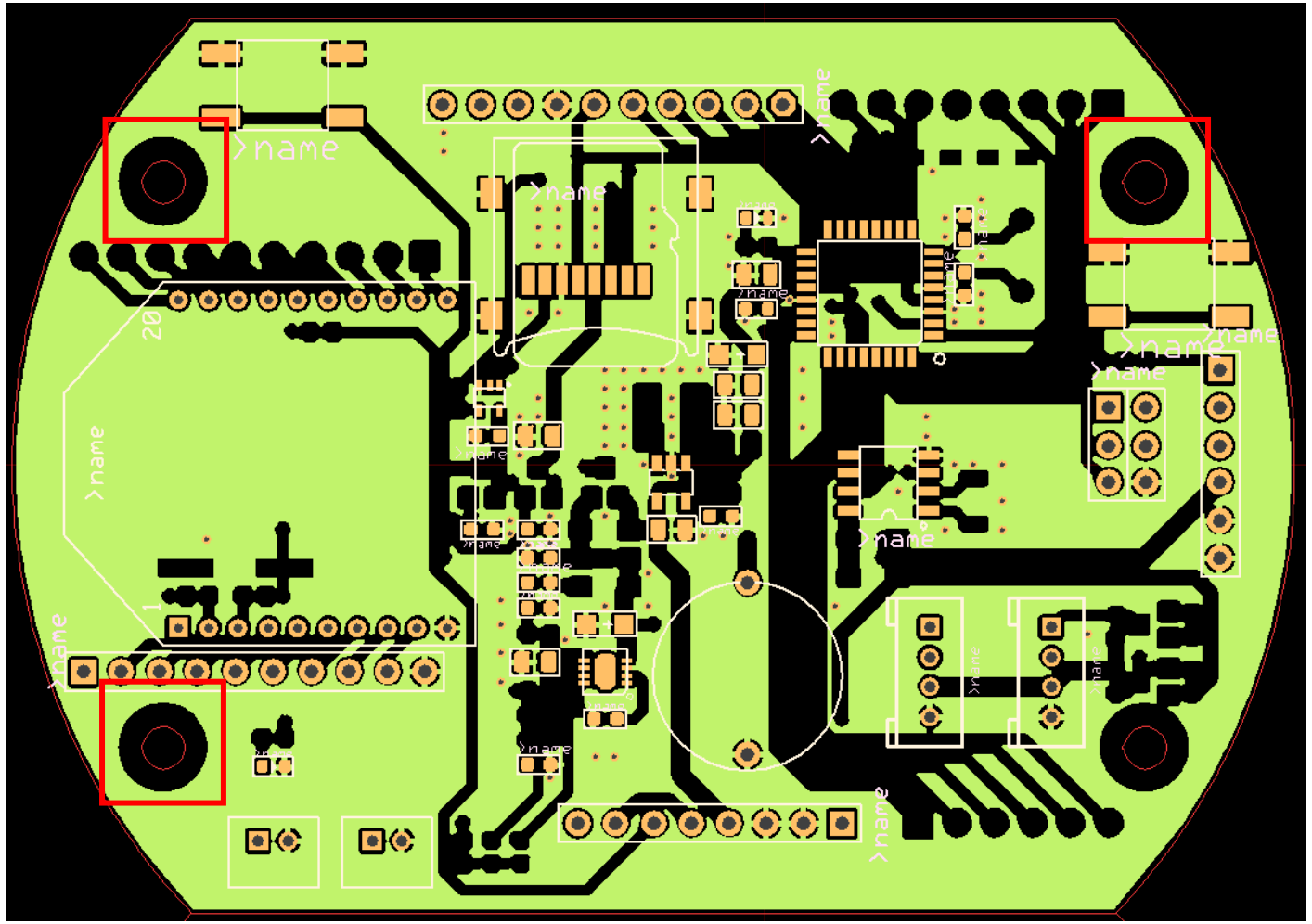
Expand Shrink Algorithm

2021. 03. 15

개발 1팀 윤서영

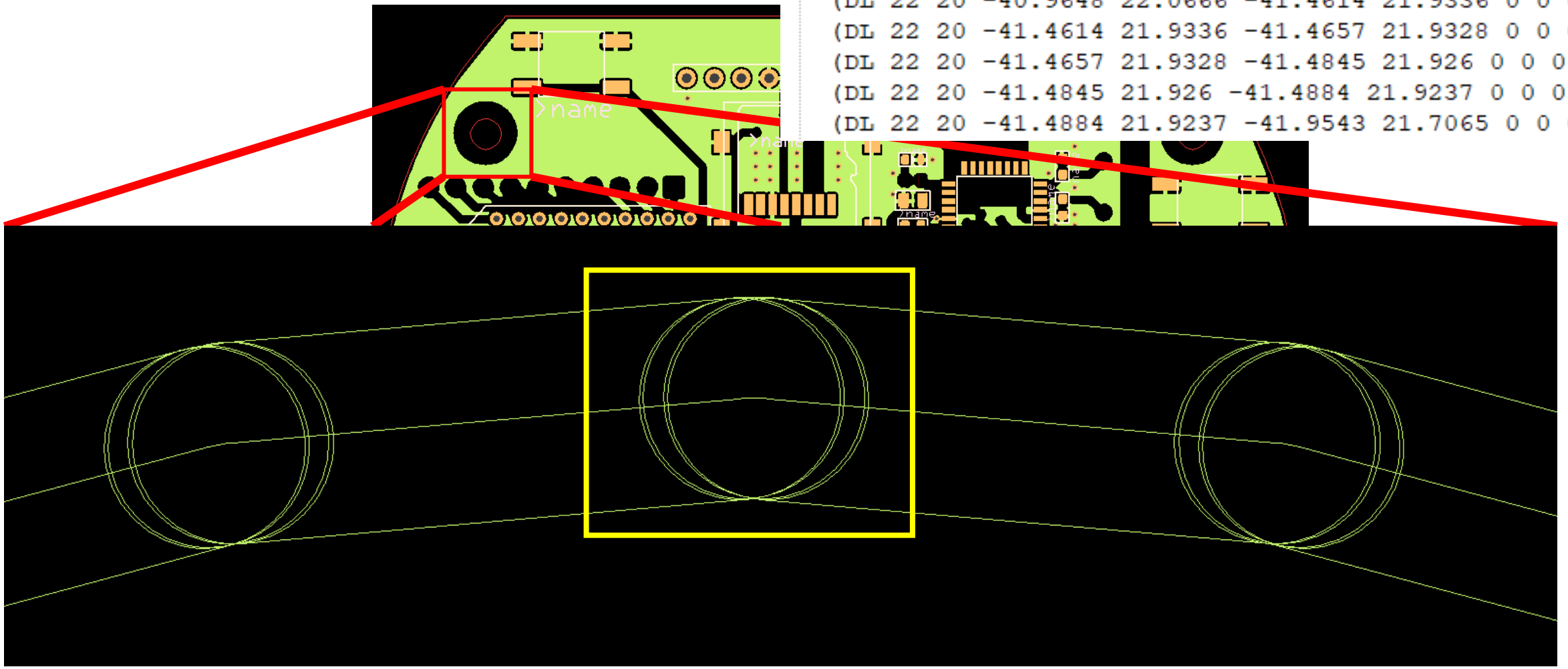
Expand(Shrink) 오류 샘플



Expand(Shrink) 오류 샘플

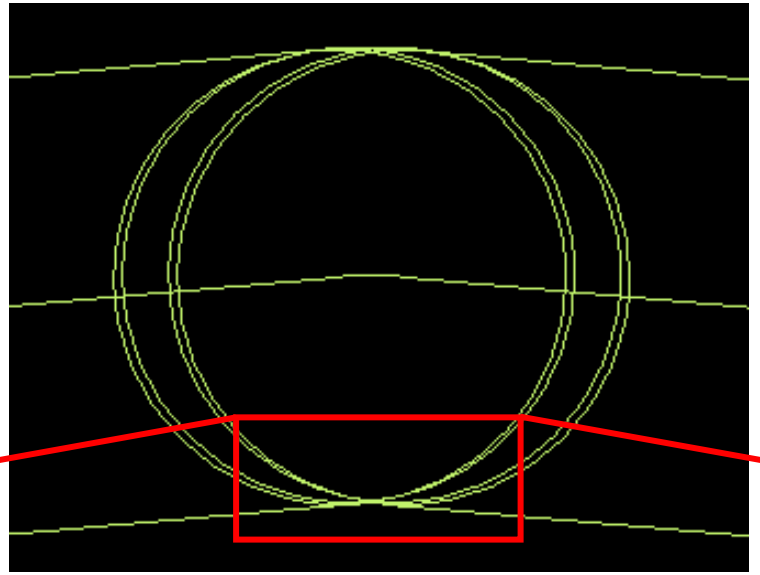
- 아래 샘플은 Polygon 내부 Line 으로 처리된 샘플

```
(POLYGON 22
    (DL 22 20 -40.4 22.1172 -40.42 22.1172 0 0 0 0 0)
    (DL 22 20 -40.42 22.1172 -40.4244 22.1164 0 0 0 0 0)
    (DL 22 20 -40.4244 22.1164 -40.9365 22.0716 0 0 0 0 0)
    (DL 22 20 -40.9365 22.0716 -40.9409 22.0716 0 0 0 0 0)
    (DL 22 20 -40.9409 22.0716 -40.9606 22.0682 0 0 0 0 0)
    (DL 22 20 -40.9606 22.0682 -40.9648 22.0666 0 0 0 0 0)
    (DL 22 20 -40.9648 22.0666 -41.4614 21.9336 0 0 0 0 0)
    (DL 22 20 -41.4614 21.9336 -41.4657 21.9328 0 0 0 0 0)
    (DL 22 20 -41.4657 21.9328 -41.4845 21.926 0 0 0 0 0)
    (DL 22 20 -41.4845 21.926 -41.4884 21.9237 0 0 0 0 0)
    (DL 22 20 -41.4884 21.9237 -41.9543 21.7065 0 0 0 0 0)
```



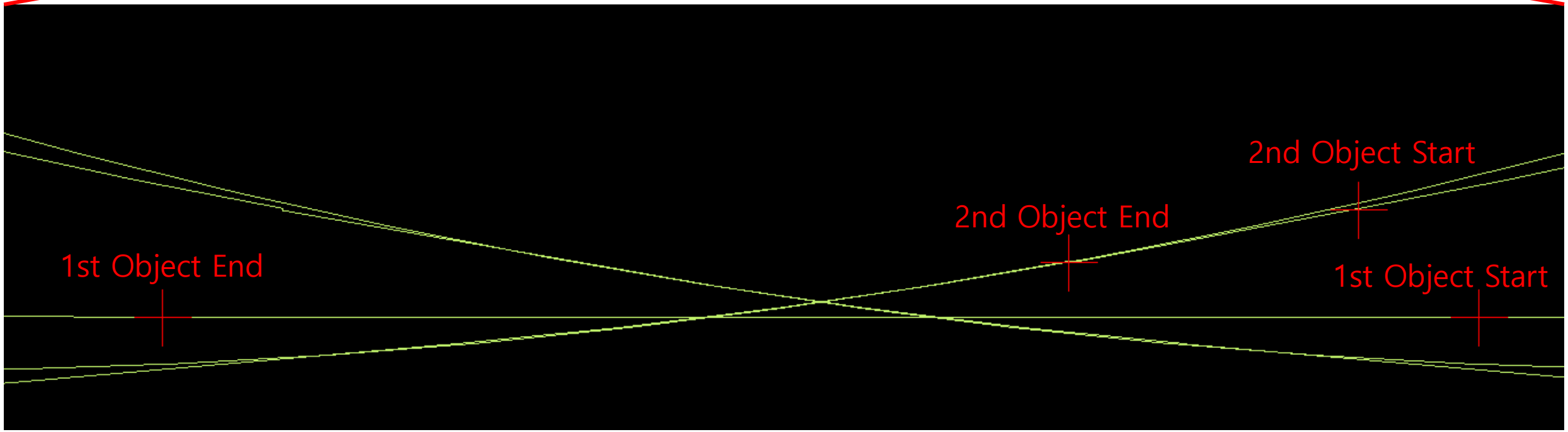
Expand(Shrink) 오류 샘플

- 접점이 존재하지 않음



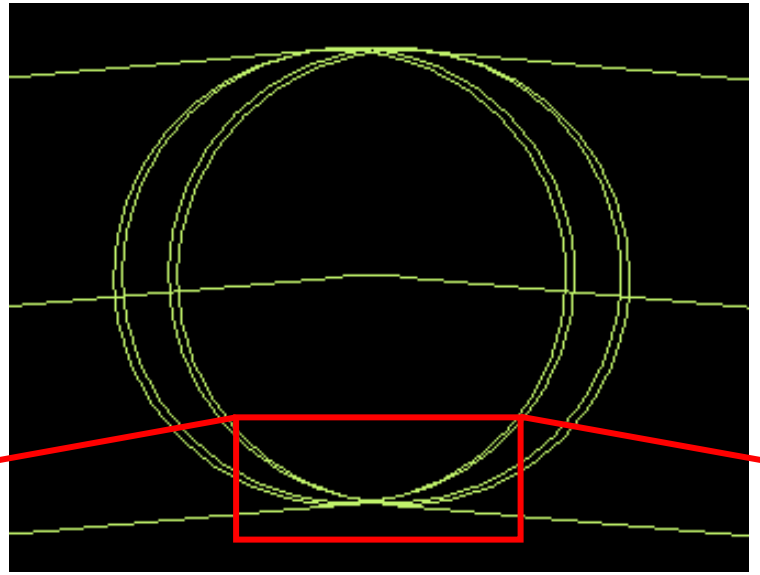
1st Object Start
-40.399999999999999 22.015600500000001
1st Object End
-40.420000000000002 22.015600500000001

2nd Object Start
-40.401825328921554 22.017239309068813
2nd Object End
-40.406225328921550 22.016439309068812



Expand(Shrink) 오류 샘플

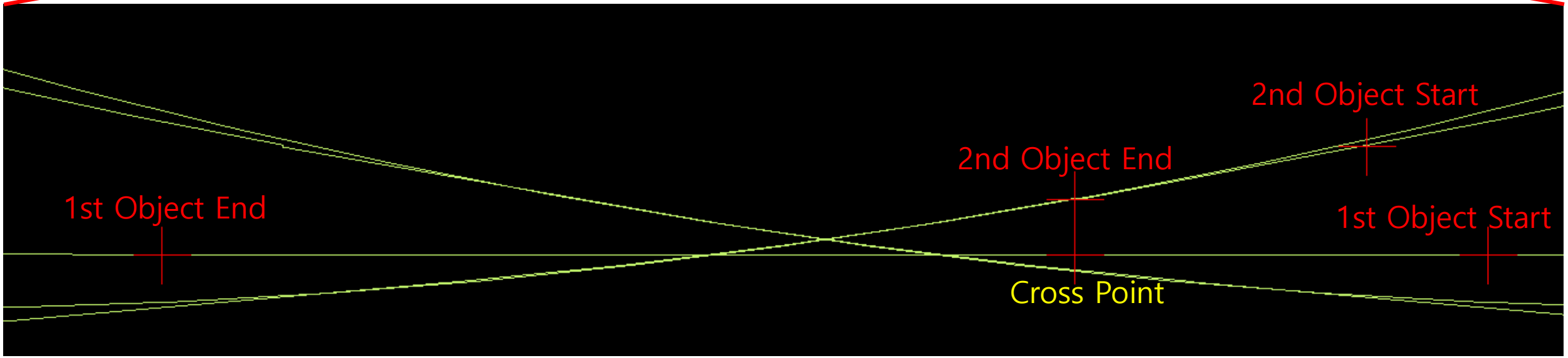
- 접점이 존재하는 결과가 나옴



1st Object Start
-40.399999999999999 22.015600500000001
1st Object End
-40.420000000000002 22.015600500000001

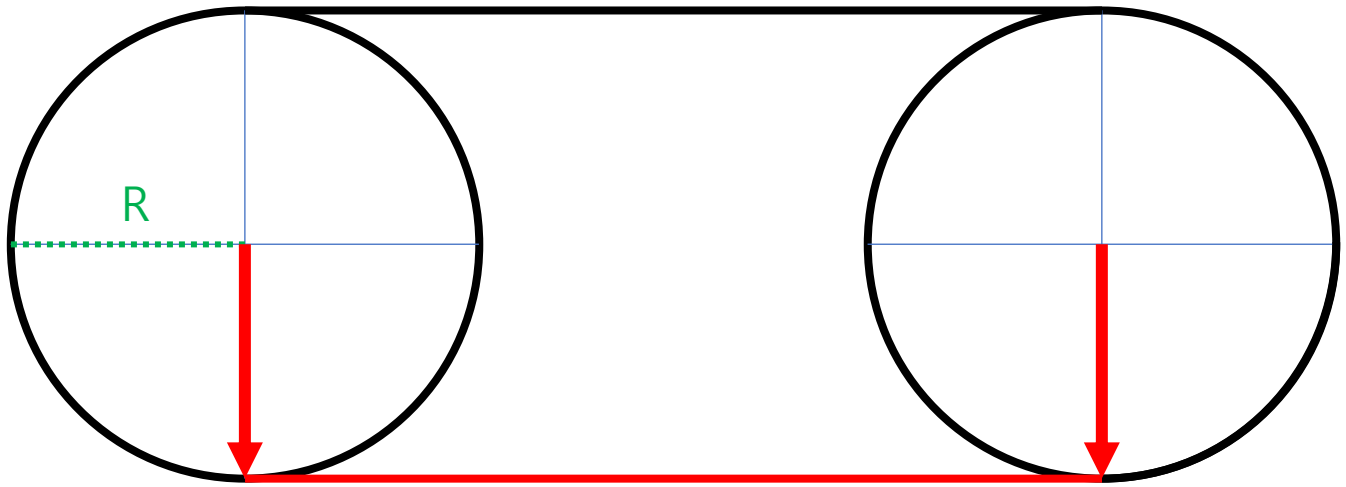
2nd Object Start
-40.401825328921554 22.017239309068813
2nd Object End
-40.406225328921550 22.016439309068812

Cross Point
-40.406225328921550 22.015600500000001



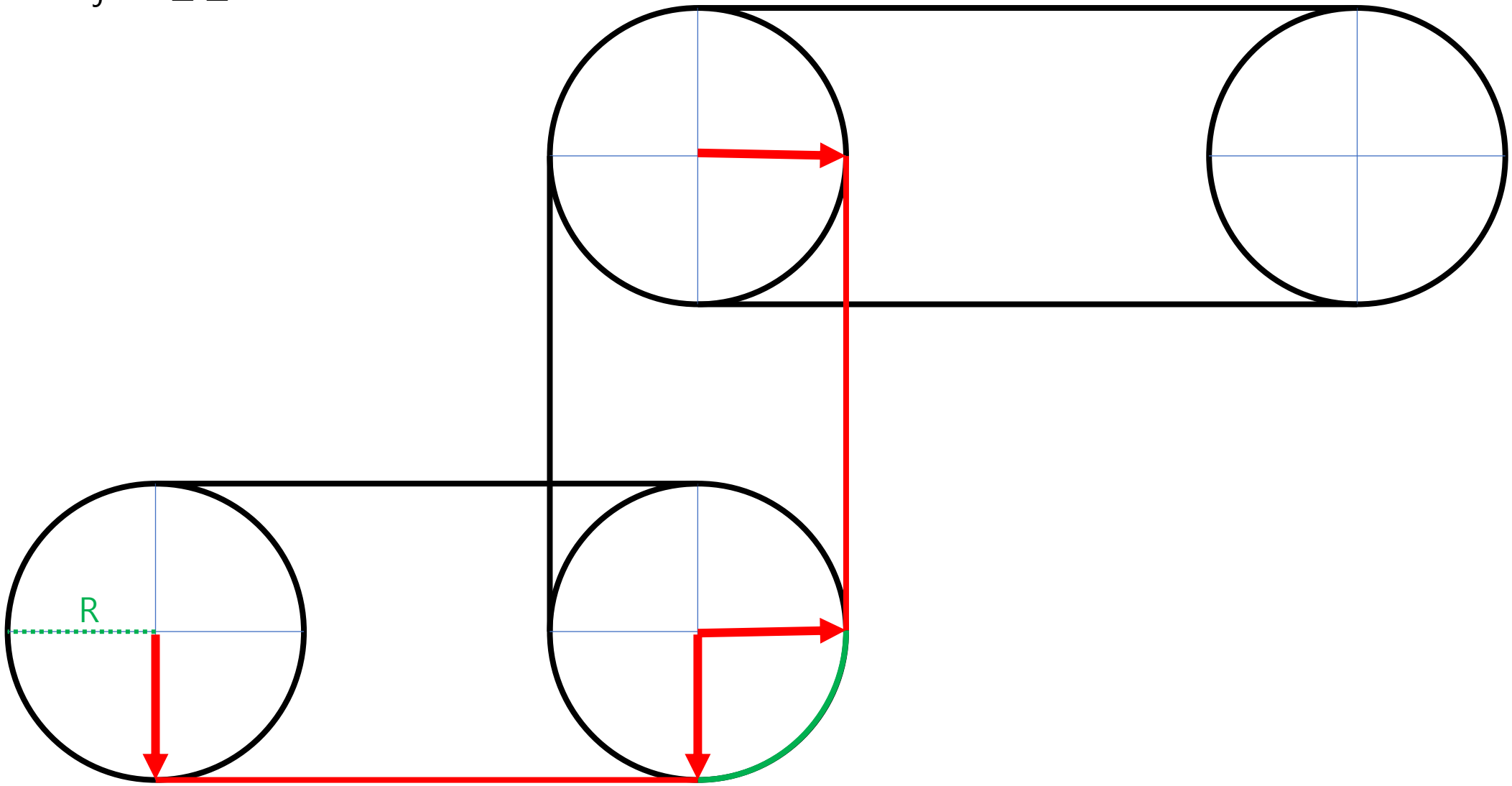
Expand(Shrink) 기본 개념

- Expand(Shrink) 방향으로 R(Turret Width)만큼 이동
- 이동된 좌표를 Start, End Point로 갖는 Object 생성(기존 Object와 동일 Type, Turret Size = 0)



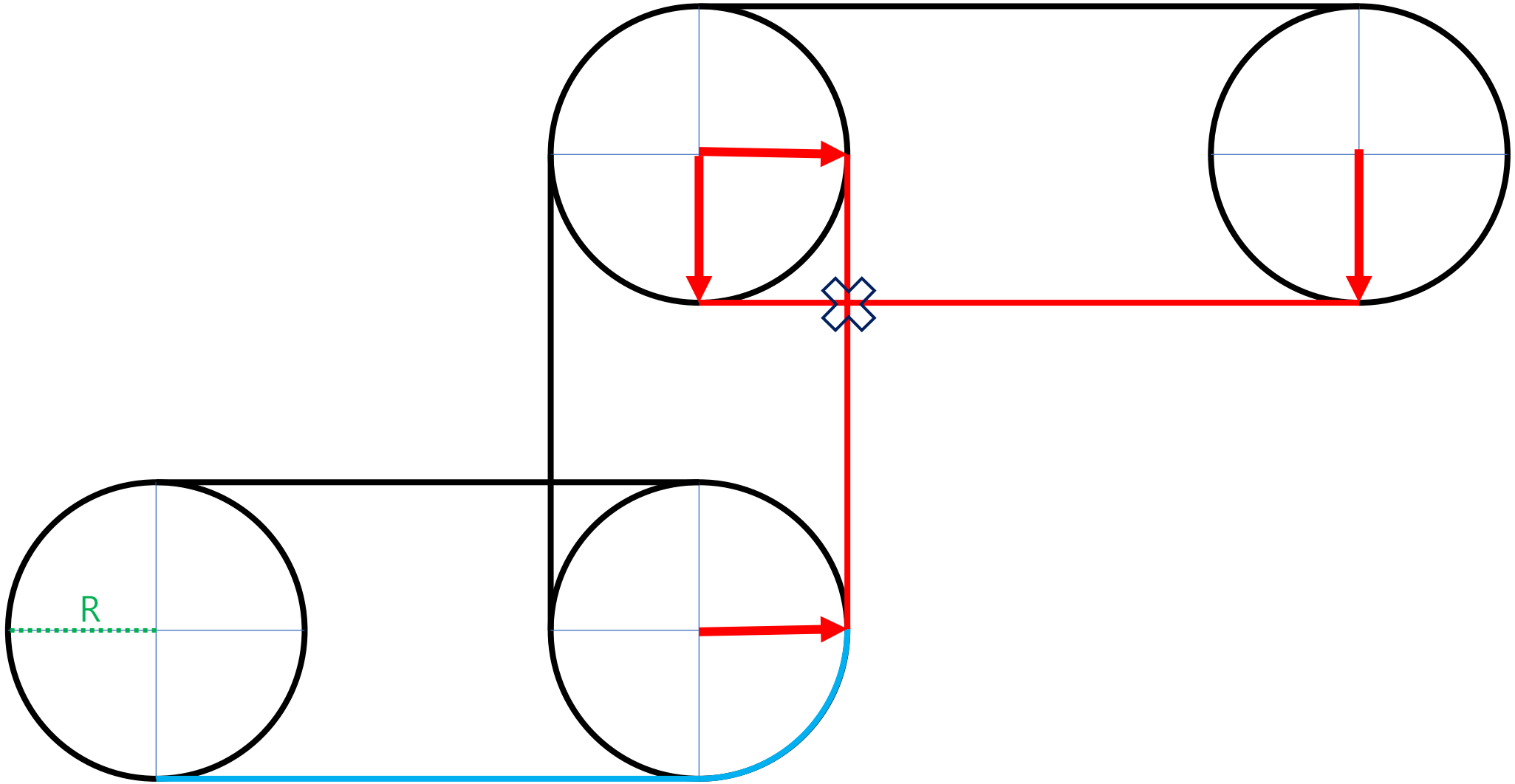
Expand(Shrink) 기본 개념

- Object 별 좌표 이동 결과를 통해 접점이 없는 경우
- Arc로 Object 연결



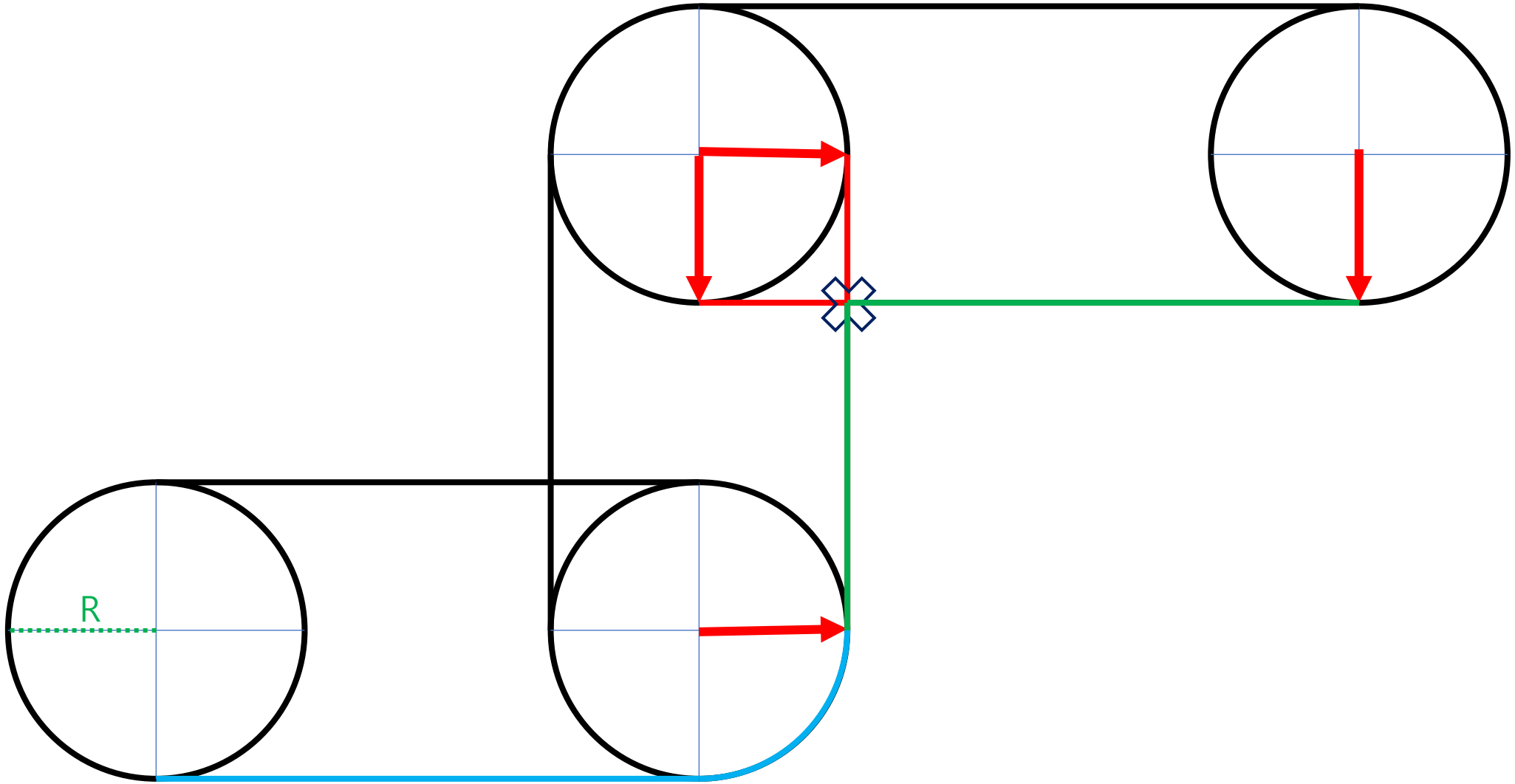
Expand(Shrink) 기본 개념

- Object 별 좌표 이동 결과를 통해 접점이 존재하는 경우
- 접점을 기준으로 Object의 Start, End Point 변경처리



Expand(Shrink) 기본 개념

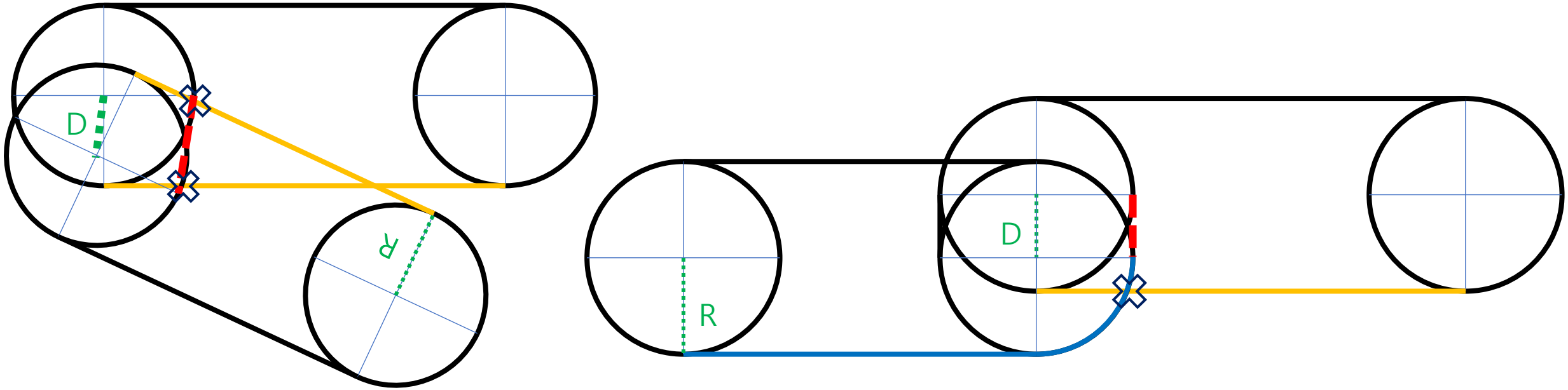
- Object 별 좌표 이동 결과를 통해 접점이 존재하는 경우
- 접점을 기준으로 Object의 Start, End Point 변경처리



Expand(Shrink) 진행 방식

1. 연결하고자 하는 Object를 Expand(Shrink)결과 Object를 모은 PolyLine과 접점 검사
   (접점 검사는 CPP_Measure Class의 Calc_MinDist 함수 이용)
2. PolyLine 중간에 접점이 있는 경우 해당 위치를 연결지점으로 지정
3. PolyLine의 마지막 Object 이전에 접점이 생긴 경우 이후 Object는 삭제
4. 연결이 가능한 Object는 PolyLine의 ObjectArray에 추가

* 단, 원본 Polygon의 마지막 Object의 경우 접점이후 삭제에 대해 예외처리



PolyLine으로 만들어 접점 검사 이후 삭제가 필요한 경우