```
<< "/home/ester/Documents/mathematica/OneCycles/OneCyclesNew.m"
```

```
meight := MC[Pp[6, 1], Pp[2, 5], Pm[4, 7], Pm[8, 3]];
mtest1 := MC[Pp[1, 3], Pm[5, 2], Pp[8, 3], Pm[6, 4], Pp[7, 9]];
mtest2 := MC[Pp[3, 1], Pp[2, 7], Pp[10, 4], Pp[9, 5], Pp[6, 8], Pp[11, 7], Pp[10, 9]];
mtest3 := MC[Pp[3, 1], Pp[2, 7], Pp[10, 4],
   Pp[9, 5], Pp[6, 8], Pp[11, 7], Pp[10, 9], Pp[11, 2], Pp[4, 9]];
mtest4 := MC[Pp[3, 1], Pp[2, 7], Pp[10, 4], Pp[9, 5], Pp[6, 8],
   Pp[11, 7], Pp[10, 9], Pp[11, 2], Pp[9, 4]];
mtest5 := MC[Pp[1, 3], Pp[2, 7], Pp[10, 4], Pp[9, 5], Pp[8, 6],
   Pp[11, 7], Pp[10, 9], Pp[11, 2], Pp[9, 4]];
```

```
FindSingR3[mtest4]
```

```
{{{2, 7, 11}, {4, 9, 10}}, {R1, L2}}
```

```
mtest2
```

```
MC[Pp[3, 1], Pp[2, 7], Pp[10, 4], Pp[9, 5], Pp[6, 8], Pp[11, 7], Pp[10, 9]]
```

```
Crossings[mtest2]
```

```
11
```

```
CrossingSign[mc_MC, p1_, p2_] := Module[
   {i, outp},
   outp = 0;
   i = 0;
   While[i++ < Length[mc], (*Print[i,mc[[i]][[1]]];*)
    If[mc[[i]][[1]] === p1 && mc[[i]][[2]] === p2 ||
       mc[[i]][[1]] === p2 && mc[[i]][[2]] === p1,
      If[Head[mc[[i]]] === Pp, outp = 1;
        Break[];, outp = -1; Break[];];
     ];
   ];
   outp
  ];
```

```
CrossingSign[mtest1, 2, 5]
```

```
-1
```

```
Head[mtest1[[1]]] === Pp
```

```
True
```

```
FindSingR3[mtest4]
```

```
Position in mc: (7,9,3), l,m,t:(10,9,4), global L2 local 1
```

```
Position in mc: (8,2,6), l,m,t:(11,2,7), global R1 local 1
```

```
FindR2[mtest5]
```

```
parallel 9,5,10,4
```

```
parallel 8,6,9,5
```

```
FindR3[mc_MC] := Module[
    {i, j, k},
    i = 0;
    While[i++ < Length[mc],
     j = i;
     While[j++ < Length[mc],
      k = j;
      While[k++ < Length[mc],
       Which[mc[[i]][[1]] + 1 == mc[[j]][[1]],
        Which[mc[[i]][[2]] + 1 == mc[[k]][[1]],
         Which[mc[[j]][[2]] + 1 == mc[[k]][[2]], Print["ij-ik-jk"];,
          mc[[j]][[2]] - 1 == mc[[k]][[2]], Print["ij-ik-kj"];], (*end which*)
         mc[[i]][[2]] - 1 == mc[[k]][[1]],
         Which[mc[[j]][[2]] + 1 == mc[[k]][[2]], Print["ij-ki-jk"];,
          mc[[j]][[2]] - 1 == mc[[k]][[2]], Print["ij-ki-kj"];](*end which*)
        ], (*end which*)
        mc[[i]][[1]] - 1 == mc[[j]][[1]],
        Which[mc[[i]][[2]] + 1 == mc[[k]][[1]],
         Which[mc[[j]][[2]] + 1 == mc[[k]][[2]], Print["ji-ik-jk"];,
          mc[[j]][[2]] - 1 == mc[[k]][[2]], Print["ji-ik-kj"];], (*end which*)
         mc[[i]][[2]] - 1 == mc[[k]][[1]],
         Which[mc[[j]][[2]] + 1 == mc[[k]][[2]], Print["ji-ki-jk"];,
          mc[[j]][[2]] - 1 == mc[[k]][[2]], Print["ji-ki-kj"];](*end which*)
        ](*end which*)
       ](*end which*)
       (*If[CommonElement[mc[[i]],mc[[j]]] , Print[i,j],]*)

      ](*while k*)
     ](*while j*)
    ](*while i*)
   ]; (* only for all forward!*)
```

```
FindR3[mtest1]
```

```
CommonElement[p1_, p2_ ] :=
  If[p1[[1]] == p2[[1]] || p1[[1]] == p2[[2]] ||
    p1[[2]] == p2[[1]] || p1[[2]] == p2[[2]], True,];
```

```
Intersection @@@ {mtest3[[3]], mtest3[[7]]}
```

Intersection::normal : Nonatomic expression expected at position 1 in 10 ⋂ 4. ≫

Intersection::normal : Nonatomic expression expected at position 1 in 10 ⋂ 9. ≫

```
{10 ⋂ 4, 10 ⋂ 9}
```

```
Intersection @@ {mtest1[[1]], Pm[1, 4]}
```

Intersection::heads : Heads Pm and Pp at positions 2 and 1 are expected to be the same. ≫

```
Pp[1, 3] ⋂ Pm[1, 4]
```