COSC229 Tutorial No.4
Coomputational Geometry

7th June 2005

(1) Intersection

Let

(1):turn(s1.p1,s1.p2,s2.p1) (2):turn(s1.p1,s1.p2,s2.p2)

(3):turn(s2.p1,s2.p2,s1.p1) (4):turn(s2.p1,s2.p2,s1.p2)

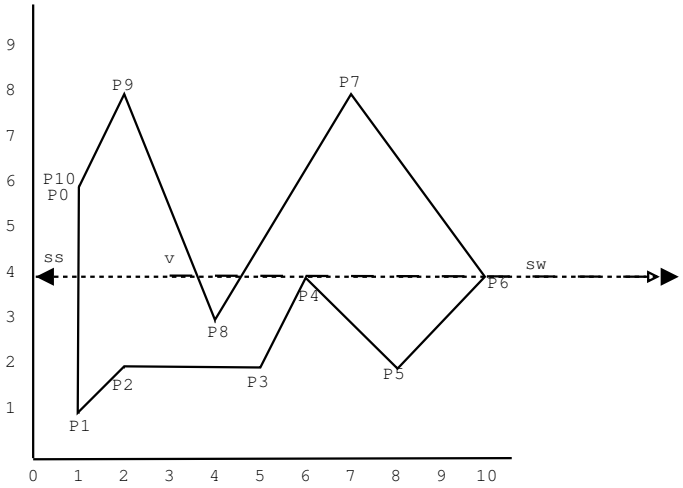
Eg	(1)	(2)	A:(1)*(2)	(3)	(4)	B:(3)*(4)	A&&B
1	-1 (case 2)	1 (case 1)	-1 (T)	1 (case 1)	-1 (case 2)	-1 (T)	T
2	-1 (case 2)	0 (case 5)	0 (T)	1 (case 1)	-1 (case 2)	-1 (T)	T
3	-1 (case 2)	-1 (case 2)	1 (F)	1 (case 1)	-1 (case 2)	-1 (T)	F
4	-1 (case 2)	0 (case 5)	0 (T)	1 (case 1)	0 (case 5)	0 (T)	T
5	-1 (case 2)	1 (case 4)	-1 (T)	1 (case 1)	1 (case 1)	1 (F)	F

(2) Inside

Refer to P.57 of your lecture note for the code.

Trace the algorithm “inside” with v=(3,4) and v=(4,4)

for P=(1,1),(2,2),(5,2),(6,4),(8,2),(10,4),(7,8),(4,3),(2,8),(1,6)



v=(3,4)

i	j	sp	!intersect(sp,sw)?	sp	i=j+1?	count
1	0	P1→P1	O	P1→P0	O	0
2	1	P2→P2	O	P2→P1	O	0
3	2	P3→P3	O	P3→P2	O	0
4	3	P4→P4	X	skip	skip	0
5	3	P5→P5	O	P5→P3	X	0
6	5	P6→P6	X	skip	skip	0
7	5	P7→P7	O	P7→P5	X	1
8	7	P8→P8	O	P8→P7	O	2
9	8	P9→P9	O	P9→P8	O	3
10	9	P10→P10	O	P10→P9	O	3

Final count=3 (odd). v is inside.

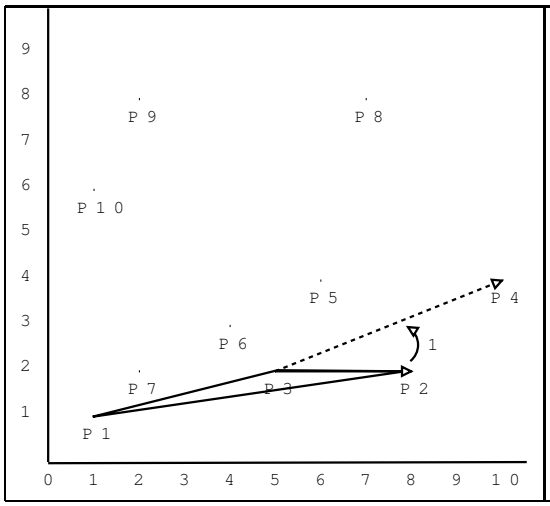
v=(4,4)

i	j	sp	!intersect(sp,sw)?	sp	i=j+1?	count
1	0	P1→P1	O	P1→P0	O	0
2	1	P2→P2	O	P2→P1	O	0
3	2	P3→P3	O	P3→P2	O	0
4	3	P4→P4	X	skip	skip	0
5	3	P5→P5	O	P5→P3	X	0
6	5	P6→P6	X	skip	skip	0
7	5	P7→P7	O	P7→P5	X	1
8	7	P8→P8	O	P8→P7	O	2
9	8	P9→P9	O	P9→P8	O	2
10	9	P10→P10	O	P10→P9	O	2

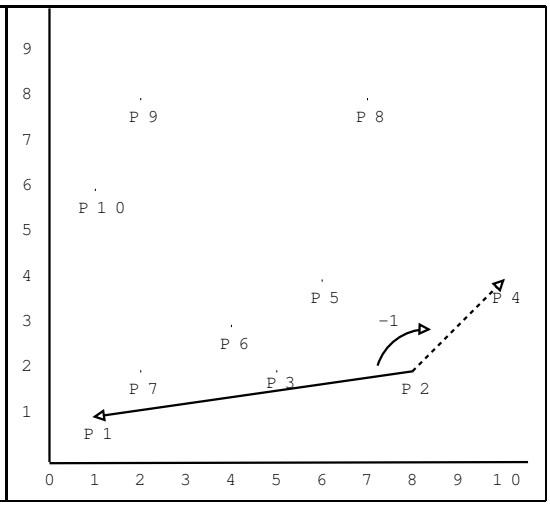
Final count=2 (even). v is NOT inside.

(3) Graham's algorithm for convex hull (p.60)

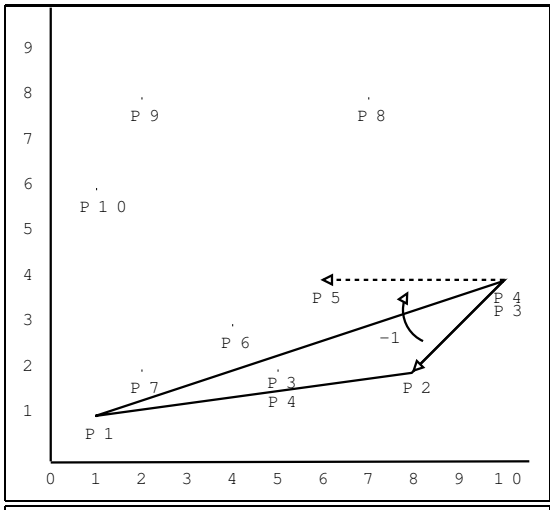
Take the lowest point (1,1) as P1 and sort other points by the angle they make with x-axis. m is the number of points included in the current convex hull. We start with a convex hull of size m=3, and start "for" loop with i=4. Each time when we decide to include a new point into the convex hull, m is increased. When we decide to discard a point from the current convex hull, m is decreased. This decision is made based on the returned value of "turn" function.



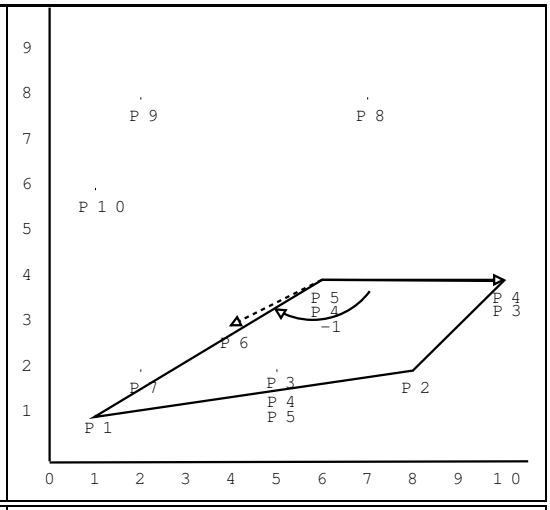
i=4. m=3.
turn(P3,P2,P4)=1. m:= 2



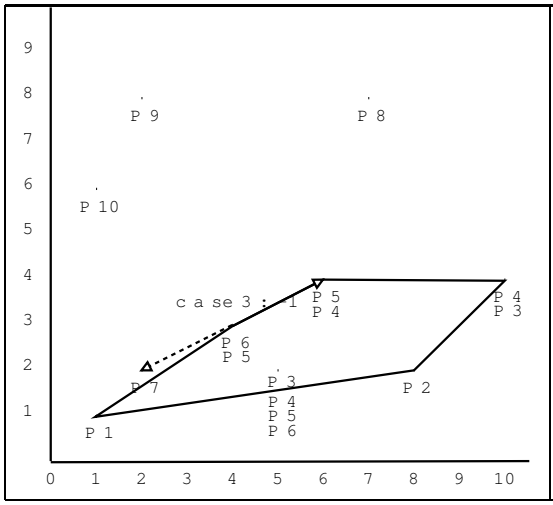
turn(P2,P1,P4)=-1. m:=3.
Swap(P3,P4)



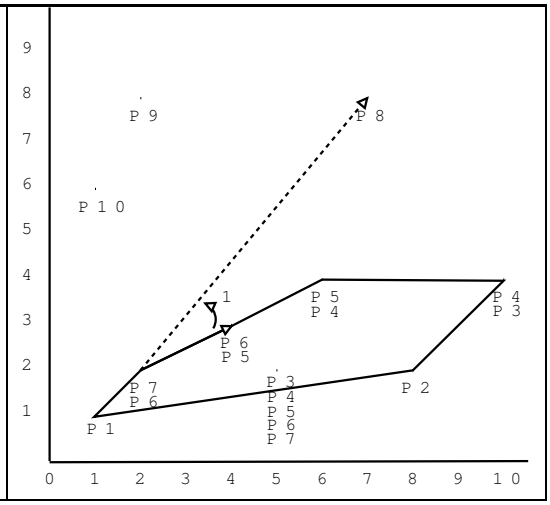
i=5. m=3.
turn(P3,P2,P5)=-1. m:= 4. Swap(P4,P5)



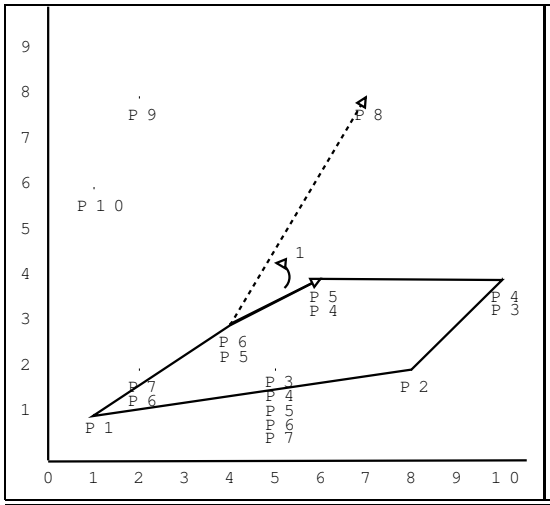
i=6. m=4.
turn(P4,P3,P6)=-1. m:=5. Swap(P5,P6)



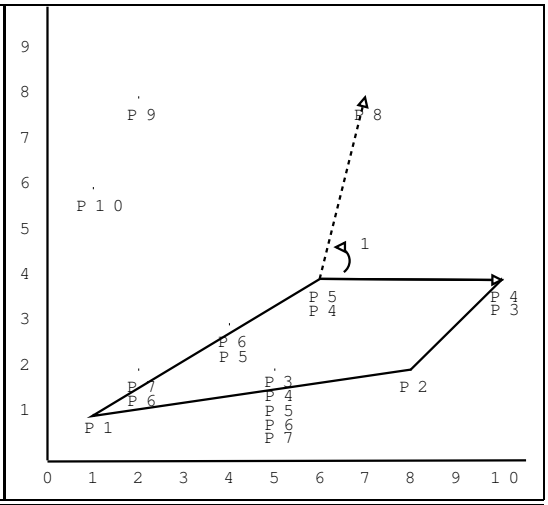
i=7. m=5.
turn(P5,P4,P7)=-1. m:= 6. Swap(P6,P7)



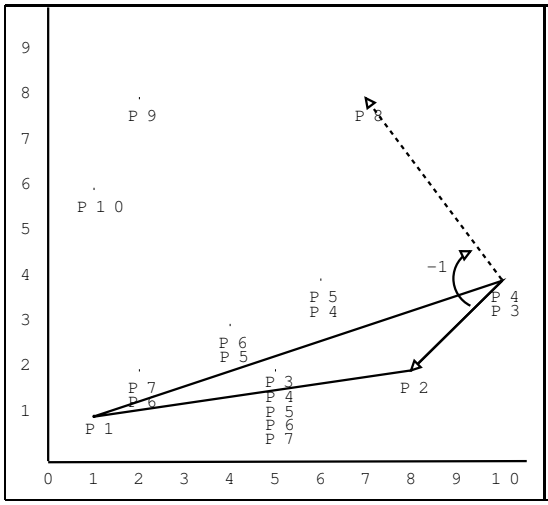
i=8. m=6.
turn(P6,P5,P8)=1. m:=5



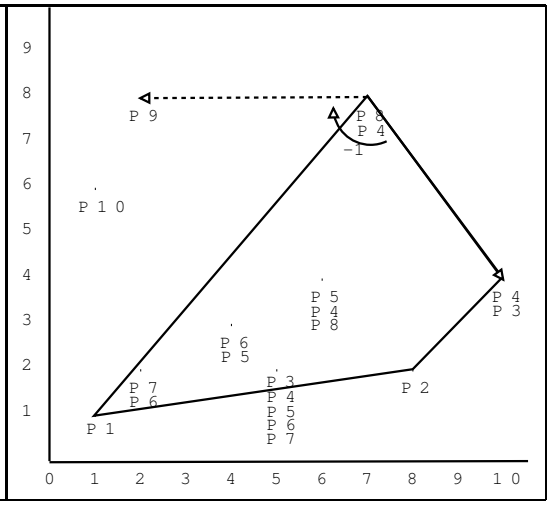
turn(P5,P4,P8)=1 m:= 4.



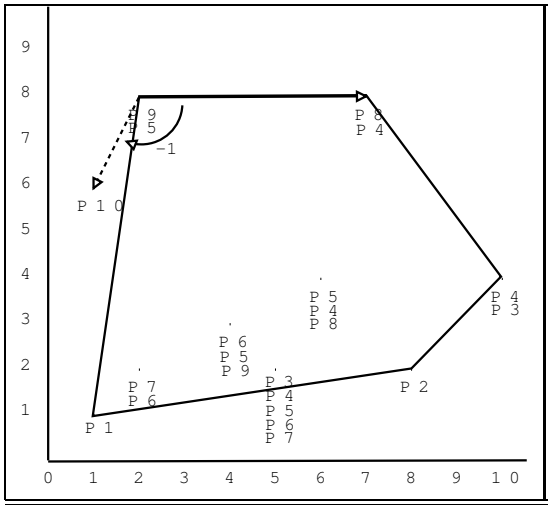
turn(P4,P3,P8)=1. m:=3



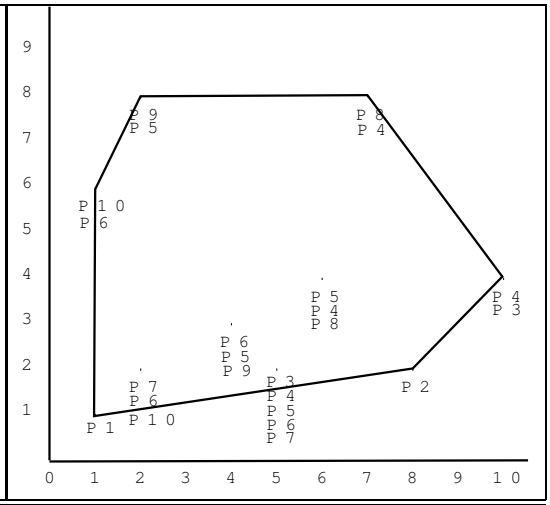
turn(P3,P2,P8)=-1. m:=4. Swap(P4,P8)



i=9. m=4.
turn(P4,P3,P9)=-1. m:=5. Swap(P5,P9)



i=10. m=5.
turn(P5,P4,P10)=-1. m:=6.
Swap(P6,P10)



Final Convex hull
including 6 points