Tutoroial-2 Solution

Answer to Q no. 1(a)

OP= Cianssprooduct (A, Q, P)

Deteromining Uppers Tangent:

$$T_1$$
-index = get_reightmost point_index $(P_nn) = 6$
 T_2 -index = get_leftmost point_index $(Q,m) = 9$

Subiteration-1: Trooversing convex hull Q

Iteration 1.2:

1-1 (10,10,10) -10

1,0 = 180 Pus 20=112.

```
if (CCW(points [i], points [i+1], Q) ==0) noturon 2
 v), if (ccw(points[i], points[i+1], Q)>082
       CCW(points[0], points[n-1], Q) ==0) peturn 2
 vi) else if (ccw(points[i], points[i+1], Q)>0 RR
            CCW(points[0], points[1], Q == 0) ractuson 2
 vii) dse if (CCW(points[i], points[i+1], Q)>0) returns.
 VIII) else rocturon O. Jatriog) no gylos
 point_in-polygon-querry (PEJ, N, QEJ, M)
i) calculate P-index = get-leftmostpoint-index (P, N).
1) rootate (P. begin(), P. begin()+Po-index, P. end()).
111) for i=0 to Q-1:
  III. a. calculate f=point-in-polygon (P,N,Q[i]).
  111. b. if (f==1) proint "Point is inside polygon".
      else if (f==2) proint "Point on boundarry of polygon".
       else proint "Point is outside polygon cos camscanner
```

Tutoroial-2 Solution

Answer to Q no. 1(a)

OP= Cianssprooduct (A, Q, P)

Deteromining Uppers Tangent:

$$T_1$$
-index = get_reightmost point_index $(P_nn) = 6$
 T_2 -index = get_leftmost point_index $(Q,m) = 9$

Subiteration-1: Trooversing convex hull Q

Iteration 1.2:

1-1 (10,10,10) -10

1,0 = 180 Pus 20=112.

Iteration-1.3:

i%m=1 T2=Q1

CCW(P6,Q1,Q2)<0

i, i%m =1!=9

· done = 0

 $T_2 = O_1$ T_2 -index=1

Traversing convex hull P

Iteration-1.1:

(3%n=6 PTI=P6

CCW(Q1, P6, P5)<0

Iteration-1.2:

(i)% n=5 T_2=P3 = x sbni_ T bas s0=

CCW(O1, P5, P4) <0

Iteration - 1.3:

(j+n)%n=4 T1=P4

CCW(Q1, Pu, P2)>0

~: (i+n) ofon = 4!=6

i. done = 1.

dono = 1 Herection-21:

190m=1 T = Q1

CCW (Ph. Q2. Q2) >0

Aeration -2.2:

10/0m=2 72=02

(CIV (Py. Q2, Q3) <0

T= 12=1001.

Corver hall P Fleraction - 2.1:

7= F = 4 90 (M)

C(M(O), R, R)>0

Ti=Py and Ti-inden=9

CS CamScanner

TI=P4 T1-index=4 Itanation-1.3; not Iteration-2: igom = 1 T2 = Q1 done = 1 COW(P.Q1.Q2)<0 Convex hull Q Iteration-2.1: i%m=1 T2=Q1 CCW(P4,Q1,Q2)>0 T=01 T2-index=1 Iteration - 2.2: i% m=2 T2=Q2 Theaversing convex hull p Peration-11: CCW (P4,Q2,Q3)<0 (800 = 6 BTE = P · i% m=2!=1 COW (Q1 , P2, P3) 20 .. done = 0 Iteraction-1.2: T2 = O2 and T2-index=2 9= 1 8= 1000 Convex hull P CCW(O10P3,P4) 20 Iteration - 2.1: (j+n)%n=4 T1=4 Henotion - 13: CCW(Q2, P4, P2)>0 (Jon) Kon = 1 12 = P.

Ti=Pu and Ti-index=4

CS CamScanner

CCM((1.1.1.1)>0

Iteration -3: Petermining Lower Jangent: done = 1 Theration-3.1: 21% m=2 T2=0+mingt20m+ts/. top= x9bmi-st T2 = P6. CCW(Q2 2P47 P2) CCW(P4,Q2,Q3)<0 12 = Qg T2 = Q2 and T2-index=2 Convex hull P Iteration -3.1: Iteration-1: (j+n)%n=4 T1=@P4 : (1) IMY XDE HOD - BF JOH HOF CCW (Q2, P4, P2)>0 Iterachion -1.1: T1=Py and T1-index=4

O=st e=mole(n+i) (-mi) So, the upper tangent contains the points Heration 1.2: Py and Q2. (iim) = 8 -12 = Q6 CCW(F2.06-02) <0

Determining Lawer Tangent:

Ti-index = get-roightmostpoint_index (P,n) = 6

Tz-index = get-leftmostpoint_index (Q,m) = 9

T1 = P6

T2 = Q9

done = 0

T2 = Q2 and T3_1310ex = 2

Iteration-1:

done=1

(1411)96n=4 T=@P.

CCW (Q2,R,R)>0

(2014(Q, 3P, 1)

20W(PL,Q,J) <0

For polyg-convex hall Q:

Iteration -1.1:

10/0m=9 (i+m)0/0m=9 T2=Q0

tong out only sins the populations the point

Iteration -1.2:

(i+m) % m = 8 T2 = Q8

CCW(P6, Q8, Q7) <0

of and Q2.

Iteration-1.3:

(i+m)%m=7 T2=Q2

CCW (P6, Q2, Q6)>0:0 Had some caises vool de

'. (i+m) %m!= 9

-- done =0

Troaversing convex hull-Pin- Thomas Deal

Iteration-1.1: j%n=6 T1=P6

CCW (Qz, P6, P6)>0

Iteration - 1.2:

j%n=0 T1=P0 = 15bni-st bno 9=1

CCW(Q2,P2,P1)<0

E. The lowers too forgent contains the imobilis.

: done = 0

· The nursed goursex proll.

Ti=Po and Ti-index = 0

Theraction -2.1

(1 Mm) = 0 TE = P.

CON(C2, P, P)<0

Heration-1.3: Iteration-2: (i+m) 76m=7 T2=O2 17 done=1 Traversing convex hull Q: 0< (0 = 0 = 1) Iteration-2.1 6 = [wgo (witi) ... (i+m) % m = 7 T2 = Qx CCW(P, O7, Q6) > Oobning tons son T2=Qx and T2-index=12xounds prisession Troaversing convex hull P: ST-IT 2- mill Iteration - 1.1: (j%n)=0 T1=Po CON (Q2 , B, B) >0 CCW(Oz, P, P1)<0 Heraction - 1.2: T1=Po and T1-index=0 infon=O TI=Po So, the lower tangent contains the points Po and & CCW (Gz, B, B, B) <0-. The meroged convex hull, 0 = 200 :. CH[]={Po,P1,P2,P4,Q2,Q3,Q4,Q5,Q2} CamScanner

Answer to Ono.1(6)

The given algorithm is not correct as the verotices of the polygon are in clockwise oroders and the algorithm will neturn false fora right turn, i.e., clockwise orientation.

The correct algorithm is as follows:

bool is-convex (Por..., Ph-1) }

for i=0,..., n-1: fore i=0, ..., n=1?)

of CPin Print & STEN STEP STORES

if (CCW(Pi oP(i+1) 90n o P(i+2) 96n) >0); nel roeturon false; 77 99

roeturon trong;

Doctoulate (Peanos product (P.P.P) 95 mealson

COW(4, P. P.)

Answer to Qno. 1 (c)

This algorithm takes a convex polygon and a querry of point as input and returns for each querry if the point is inside, outside on on the boundary of the polygon.

crossproduct (P2, P2, P3) mobile to series sall

Dealculate P1P2 = P2-P1=(P2.x-P1.x, P2.y-P1.y)

11) calculate P_1P_3 = P_3 - P_1 = (P_3: x - P_1 x, P_3.y - P_1.y)

11) calculate P.P. × P.P. = 3 (P2x-P1.2) (P3:y-P1.y)

(0< (, ence) - (P3. 2-P1. 2) (P2. y-P1. y) }

: and wanted

IV) roetveron PIP2 X PIP3. salot

CCW(P2, P2, P3)

Dealculate CP=crossproduct (P1, P2, P3)

11) roeturon CP.

get-leftmostpoint-index (points[],n)

1) set min-x = inf and min_ind = -1.

11) for i=0 to n-1:

11. a. if (points[i]. x < min_x || (points[i].x == min_x & points[i].y > points[min_ind].y)

set min-z=points[i] a and min-ind=i.

(ii) rocheron min-indo [[[Latinoa) NOO)

point-in-polygon (points [],n,Q)

i) if (ccw(points[0], points[1],Q)<011 ccw(points[0], points[n-1],Q)>0) neturn O.

ii) set start = 0 and end = n-2.

111) while (starot <= end):

111. a. calculate mid = (start+end)/2.

M.b. set i=mid

else set end=mid-1.

```
if (CCW(points [i], points [i+1], Q) ==0) noturon 2
 v), if (ccw(points[i], points[i+1], Q)>082
       CCW(points[0], points[n-1], Q) ==0) peturn 2
 vi) else if (ccw(points[i], points[i+1], Q)>0 RR
            CCW(points[0], points[1], Q == 0) ractuson 2
 vii) dse if (CCW(points[i], points[i+1], Q)>0) returns.
 VIII) else rocturon O. Jatriog) no gylos
 point_in-polygon-querry (PEJ, N, QEJ, M)
i) calculate P-index = get-leftmostpoint-index (P, N).
1) rootate (P. begin(), P. begin()+Po-index, P. end()).
111) for i=0 to Q-1:
  III. a. calculate f=point-in-polygon (P,N,Q[i]).
  111. b. if (f==1) proint "Point is inside polygon".
      else if (f==2) proint "Point on boundarry of polygon".
       else proint "Point is outside polygon cos camscanner
```

Answer to Qno.1(d)

Step-1: A=get_araea_of_simple_polygon(\$(7,1), V1(13,2), V2(14,7), V3 (10,9), V4 (6,10), V5 (5,2) ((4,6), Vx (4,3)) OP XOP = crossproduct ((0,0), (7,1), (13,2)) = cnossprooduct ((7,1), (13,2)) $= \begin{vmatrix} 7 & 13 \\ 1 & 2 \end{vmatrix} = (14-13) = 1$ OV, XOV, = croossproduct ((13,7)) = 63 Ov, ×Ov, = crossproduct (0,0)(14,7), (10,9)) = 56 OV3 XOV4 = crooss prooduct ((0,0),(10,9), (26,10)) = 46 OV, × OV, = crossprooduct((0,0), (6,10), (5,9)) = 4. Ov; XOV = croossproduct ((0,0),(5,9),(4,6)) =-6 Ov6 XOV2 = croossproduct ((0,0),(4,6),(4,3)) =-12 Ov2 × Ov0 = crossproduct ((0,0), (4,3), (7,1))=-17

-. A= 1+63+56+46+4-6-12-17/2.0=67.5

(P) I on D of evening Step-2: B=8 Iteroation-O: Was Ding dquie to some top of Step-3: B=B+get_LatticePoint_onLS ((7,1), (13,2)) = 8+9cd(12-1/2/143771)0)12-boogsecore (10x 30) · C130sspraoduc + ((7,1), (13,2)) =8+1-1 Iteration-1: = (EL-13) = | = | = | = | = | B=B+get_LatticePoint(onLS((13,2),(14,2)) = 8+ gcd (17-21,114-1316) 1 1 wbood 23000 = 200x 200 -8+1-1 Try x July = cross product ((0,0),(10,2), (36,10)) = 35 Iteration = 2(8,2), (0,0), (0,0)) + subsergesons = VO x , vo B=B+get-LatticePoint_onLS((14,7), (10,9)) -9 : A = 1011 3+56+5619-6-12-17/12.0 = 61.5

Iteration-3: B=B+get_LatticePoint_onLS ((10,9), (6,10)) =9+gcd(110-91,16-101)-1 -9+1-1 -9 Iteration-4: B=B+get-LatticePoint-onLS((6,10), (5,9)) =9+gcd(19-101, 15-61)-1 -9+1-1 --9 Iteration-5: B=B+get-LatticePoint_onLS((5,9), (4,6)) =9+get gcd(16-91,14-51)-1 =9+1-1 Iteration-6: B=B+get_Lattice Point_on LS ((4,6), (4,3)) =9+gcd(13-61, (4-41)-1

-9+3-1

=11

Iteration-7:

B=B+get-LatticePoint-onLS ((4,3), (7,1))

=11+gcd(11-31,17-41)-21-10-011) hop 70.

=11+1-1

-11

Step-4: T=A-B (01.60) 21 no-Inidos Hal-top 18 = 8 1-(12-31,601-61) boote=

=67.5-11+1

=63

(= B+ get_lattize Birt on LS((5,9), (4,6)) 1-(12-21-14-21)-I

Meroation-3;

Herochon-4:

1-1-6-

terestion - 5:

3-11:401

Mart : Heelfish. col ((1.6) - (4.3))

1-- [13-- 1]