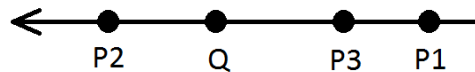
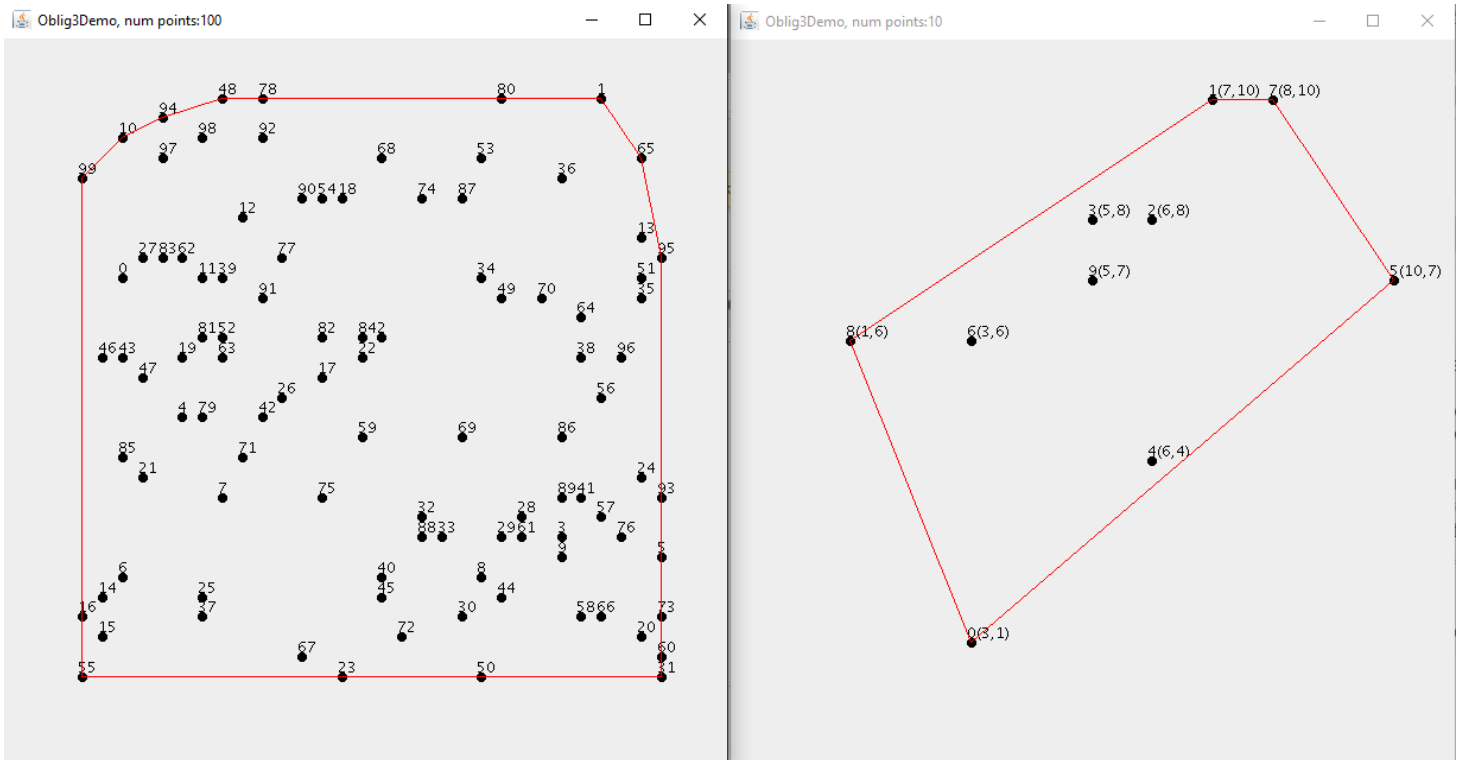


Som du har nevnt i tilbakemeldingen, gjorde jeg en feil med rekkefølgen når det er flere punkter på en linje. Etter analysering av problemet, synes jeg at feilen skjedde i steg 3B av algoritmen. Problemet er at i steg 3B har jeg ikke sjekket om et punkt Q (på linjen $p_1 \rightarrow p_2$) faktisk ligger mellom de to punktene. For eksempel, anta nå at vi har p_1 , p_2 , p_3 , og Q på samme linje, hvor p_1 og p_2 spesifiserer denne linjen, og p_3 er furthest punkt (se på figuren nedenfor). Vi lar også `list0` være en liste som inneholder alle punkter mellom (eller over) linjen $p_1 \rightarrow p_3$, og `list1` være en liste som inneholder alle punkter mellom (eller over) linjen $p_3 \rightarrow p_2$. Det betyr at `step3B()` metoden blir kallet slik: `step3B(p1,p2,p3,ptSets,list0,list1)`. Men hvis vi ikke sjekker om Q er mellom $p_1 \rightarrow p_3$, kan dette punktet ende opp i `list0`, som er galt.



For å fikse dette problemet for begge sekvensielle og parallelle versjonen, har jeg lagt til en ny metode som har signatur: `boolean isBetween(p3,p1,p2)`. Denne metoden skal sjekke om p_3 er faktisk mellom p_1 og p_2 . I tillegg har jeg også endret `step3B()` metoden slik at den kaller på `isBetween()` hver gang de tre punktene ligger på samme linje. Dette fungerer helt fint for begge den sekvensielle og parallelle løsningen. Jeg har testet koden med $n = 100$ og $n = 10$, og de punktene på kohyllen kommer ut i riktig rekkefølgen nå (se `printout.txt`).



Jeg har også funnet en annen bug i koden min. I `step2()` metoden i mitt første førsøk, kan en punkt *enten* være i listen `left` eller `right` i forhold til linjen `maxx→minx`. Anta nå at for $n = 100$, har vi valgt punkt 31 som `maxx`, og punkt 55 som `minx` (se på bilde). Det betyr at *alle* punktene ligger enten på eller over linjen `maxx→minx`, og *ingen* av de punktene ligger under denne linjen. Da vil *alle* punktene bli lagt inn i `right` listen, mens *ingen* av de punktene blir lagt inn i `left` listen. Dette her betyr at punkt 23 og punkt 50 ikke vil bli tatt med i kohyllen. For å fikse dette problemet, har jeg endret koden min i `step2()` og monitorens konstruktør, slik at hver gang vi vet at et punkt ligger på linjen `maxx→minx`, så legger vi dette punktet inn i begge `left` og `right` list. Da blir alt riktig.