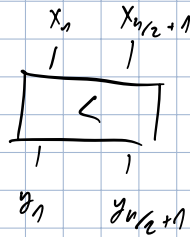
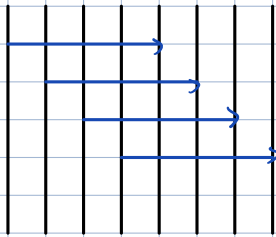
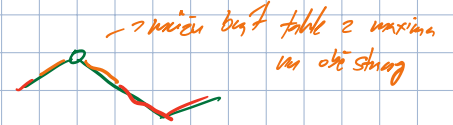


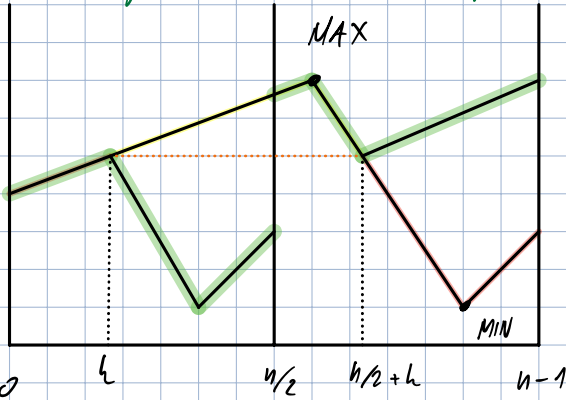
Sestrojení Separatoru pro hloubku  $O(1)$ , velikost  $O(n)$ :



$n/2$  největších prvků:



členění jsou dvě nové bitonické posl.



hova } tvoří souvislý úsek  $x_l, x_{l+1}, \dots, x_{l+n/2-1}$  → určité bitonické  
 údolí } souvislý úsek zbylých prvků  $x_{n/2+h}, x_{n/2+h+1}, \dots, x_{n+h-1}$  → křivkuje na konci z obou stran  
 BUVNO k těm v levé polovině

Pro  $i < l$ : komparátor neprobíhá

Pro  $i \geq l$ : komparátor probíhá

Opět máme bitonickou posloupnost:

- nalevo vznikla zrotovaná údolí
- napravo vznikla zrotovaná hova

Stávajícími body se dají dvě seřazené posl., jednu otočit, z toho vznikne bit. posl. a tu umí seřadit.

MergeSort:

hloubka:  $\Theta(\log^2 n)$

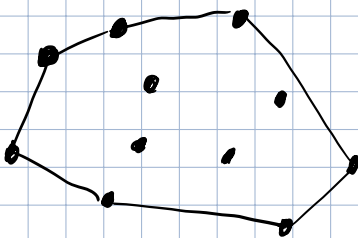
velikost:  $\Theta(n \log n)$

### Geometrické alg:

Problém ohraničení bodů:

Rozhodně bude ohraničení konvexní.

Dároveň ideální plot je mnohoúhelník, čímž ho uzavírá.



Def: Konvexní obal:

Ukončené množiny bodů  $x_1, \dots, x_n$

je konvexní mnohoúhelník s vrcholy v některých  $x_i$ .

Předpokládáme, že všechny  $x$ -ové souřadnice jsou různé.

Můžeme postupovat zleva doprava po bodech a dělat samostatně horní a dolní obálky, se společným začátkem a koncem v jedné srovnávané lince. Horní jde vždy doprava, dolní vždy doleva.

Alg:

1) Seřídíme body podle  $x \rightarrow x_1 \dots x_n$  }  $O(n \log n)$

2)  $H = (x_1), D = (x_1)$

3) Pro  $i = 2 \dots n$ :

4) Dokud  $|H| \geq 2$  &  $|H[-2]H[-1]x_i$  sotáčí obleva

5) Odstraníme z  $H$  poslední prvek

6) Přidáme  $x_i$  na konec H

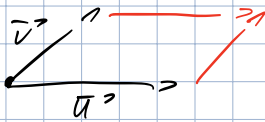
7)  $\otimes$ , doprava,  $H \rightarrow D$

$O(n)$

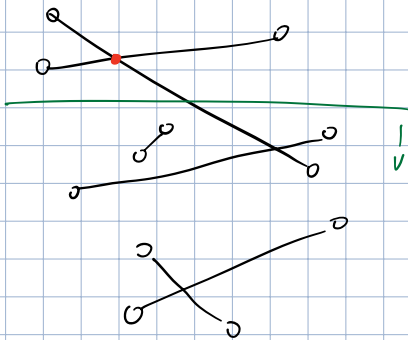
Jak řešit body ve stejném  $x$ ?

Vrcholy seřídíme lexicograficky, nikoliv jen podle  $x$ .

Pak ke každému bodu  $u$   $x$  přidám postupně.



$\begin{vmatrix} u_1 & u_2 \\ v_1 & v_2 \end{vmatrix}$  - znaménko odpovídá "otáčen" dolů / doprava



Uchvatí - průsečíky (tzv. dům dynamický)

začátek úseček   konec úseček

$\odot$  Těsně před průsečíkem se stávají úsečky sousedními

Průřez:

1) průřez =  $\emptyset$

2) kalendář = začátky a konce úseček }  $O(\log n)$

- úsečky protáčí samotné průřezem.

- přidáním do něj průřezů, na kterých jsou stápl na začátku, odebráním ty, kterých jsou stápl na konci.

3) Dokud kalendář  $\neq \emptyset$ :

Kalendář uchvatí:

4) Smažeme nejbližší uchvatí z kalendáře

- všechny budoucí uchvatí seřazení podle  $y$ .

zач. / kon. / průsečík.  
seřadím úsečky do průřezu / smažu úsečky z průřezu / přidám průsečík

zач. / kon. / průsečík.  
seřadím úsečky do průřezu / smažu úsečky z průřezu / přidám průsečík

zач. / kon. / průsečík.  
seřadím úsečky do průřezu / smažu úsečky z průřezu / přidám průsečík

Hmáčením sousednost, přehlíží průsečíkovan uchvatí.

Uchvatí spouze  $O(\log n)$

# průsečíků :=  $p$    # úseček :=  $n$

Kalendář: Binární vyhledávací strom

$\leq 3n$  uzelů  $\Rightarrow O(\log n)$  na operaci

Průřez: BTS s úsečkami jako listy

$\leq n$  pruhů  $\Rightarrow O(\log n)$  na operaci

Celkem složitost:  $O((n+p) \log n)$