

УНИВЕРЗИТЕТ У БЕОГРАДУ
МАТЕМАТИЧКИ ФАКУЛТЕТ

Геометрија И–смер

део 9: Полигони

Тијана Шукиловић

3. децембар 2023.

Полигонска линија и полигон

Дефиниција 1.1

Полигонска линија $A_0 \dots A_{n-1} A_n$ је унија дужи $A_0 A_1, \dots, A_{n-1} A_n$ које називамо **ивице** полигонске линије. Тачке A_0, \dots, A_n називају се **темена** полигонске линије.

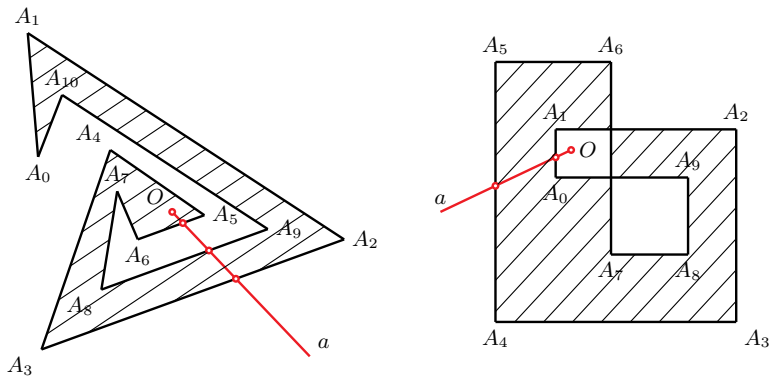
Полигонска линија и полигон

Дефиниција 1.1

Полигонска линија $A_0 \dots A_{n-1} A_n$ је унија дужи $A_0 A_1, \dots, A_{n-1} A_n$ које називамо **ивице** полигонске линије. Тачке A_0, \dots, A_n називају се **темена** полигонске линије.

- затворена полигонска линија = **полигон**
- суседна темена/ивице
- прост/сложен полигон
- дијагонала полигона
- унутрашња дијагонала полигона

Унутршњост полигона



Слика 1: Унутрашњост простог и сложеног полигона

Површина простог полигона

- Оријентисана површина полигона $P(A_0A_1 \dots A_{n-1})$

Површина простог полигона

- Оријентисана површина полигона $P(A_0A_1 \dots A_{n-1})$

Теорема 1.1

За прост полигон $A_0A_1 \dots A_{n-1}$ и произвољну тачку равни A важи:

$$P(A_0, A_1, \dots, A_{n-1}) = P(A, A_0, A_1) + \dots + P(A, A_{n-1}, A_0).$$

Површина простог полигона

- Оријентисана површина полигона $P(A_0A_1 \dots A_{n-1})$

Теорема 1.1

За прост полигон $A_0A_1 \dots A_{n-1}$ и произвољну тачку равни A важи:

$$P(A_0, A_1, \dots, A_{n-1}) = P(A, A_0, A_1) + \dots + P(A, A_{n-1}, A_0).$$

$$\begin{aligned} P(A_0, A_1, \dots, A_{n-1}) &= \frac{1}{2} \sum_{k=0}^{n-1} (x_k y_{k+1} - x_{k+1} y_k) \\ &= \frac{1}{2} \sum_{k=0}^{n-1} x_k (y_{k+1} - y_{k-1}) \end{aligned}$$

Пример

Пример 1

У равни су дате тачке $P_0 = (1, -3)$, $P_1 = (2, -2)$,
 $P_2 = (-1, 2)$, $P_3 = (4, -1)$, $P_4 = (0, 3)$.

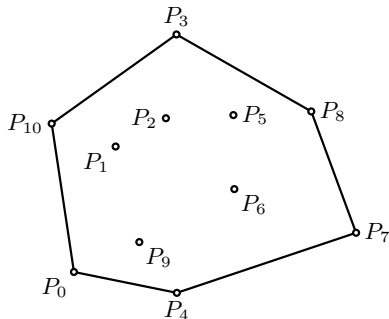
Испитати да ли је полигон $P_0P_1P_2P_3P_4$ прост.

Ако није, сортирати тачке P_0, \dots, P_4 тако да полигон буде прост.

Израчунати површину тако добијеног простог полигона.

Конвексни омотач

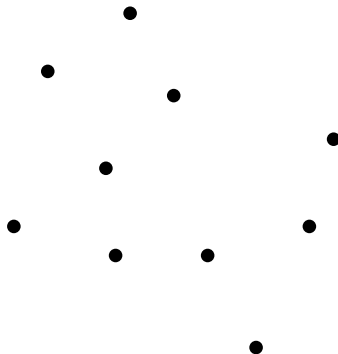
- КОНВЕКСАН ЛИК
- КОНВЕКСАН ОМОТАЧ СКУПА ТАЧАКА



Слика 2: Пример конвексног омотача скупа од 11 тачака

Алгоритам „ивичне дужи“ (спори алгоритам)

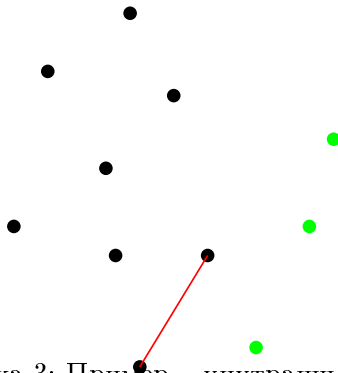
- Временска сложеност $O(n^3)$



Слика 3: Пример – 11 тачака

Алгоритам „ивичне дужи“ (спори алгоритам)

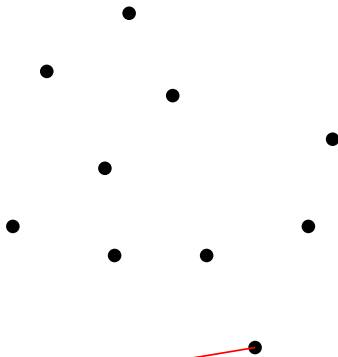
- Временска сложеност $O(n^3)$



Слика 3: Пример – унутрашња дуж

Алгоритам „ивичне дужи“ (спори алгоритам)

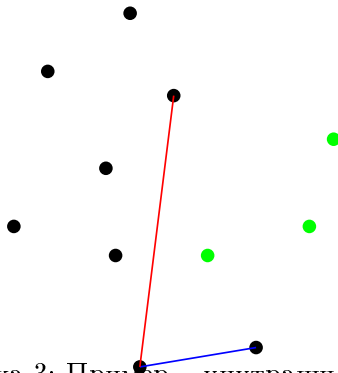
- Временска сложеност $O(n^3)$



Слика 3: Пример – ивична дуж

Алгоритам „ивичне дужи“ (спори алгоритам)

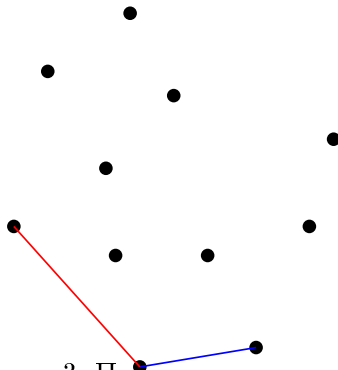
- Временска сложеност $O(n^3)$



Слика 3: Пример – унутрашња дуж

Алгоритам „ивичне дужи“ (спори алгоритам)

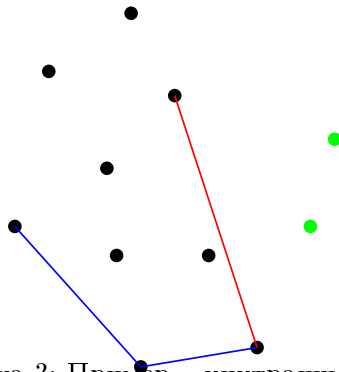
- Временска сложеност $O(n^3)$



Слика 3: Пример – ивична дуж

Алгоритам „ивичне дужи“ (спори алгоритам)

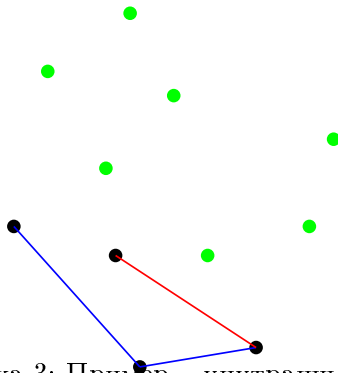
- Временска сложеност $O(n^3)$



Слика 3: Пример – унутрашња дуж

Алгоритам „ивичне дужи“ (спори алгоритам)

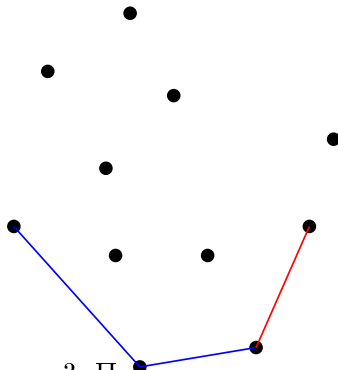
- Временска сложеност $O(n^3)$



Слика 3: Пример – унутрашња дуж

Алгоритам „ивичне дужи“ (спори алгоритам)

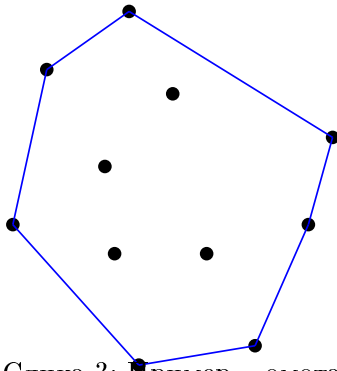
- Временска сложеност $O(n^3)$



Слика 3: Пример – ивична дуж

Алгоритам „ивичне дужи“ (спори алгоритам)

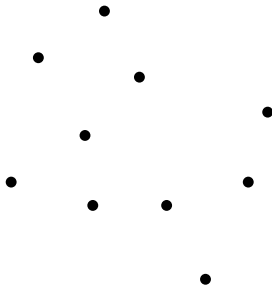
- Временска сложеност $O(n^3)$



Слика 3: Пример – омотач

Алгоритам „паковање поклона” (gift wrap)

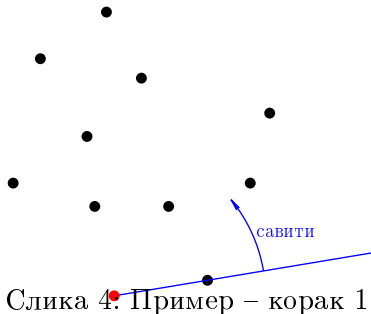
- Временска сложеност $O(n^2)$



Слика 4: Пример – P_0 = најнижа (крајња десна) тачка

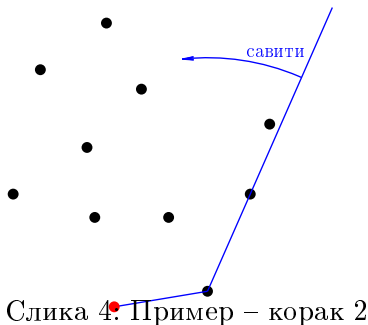
Алгоритам „паковање поклона” (gift wrap)

- Временска сложеност $O(n^2)$



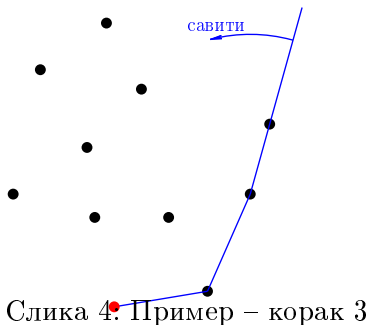
Алгоритам „паковање поклона” (gift wrap)

- Временска сложеност $O(n^2)$



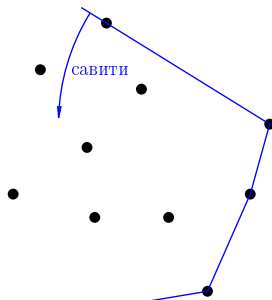
Алгоритам „паковање поклона” (gift wrap)

- Временска сложеност $O(n^2)$



Алгоритам „паковање поклона” (gift wrap)

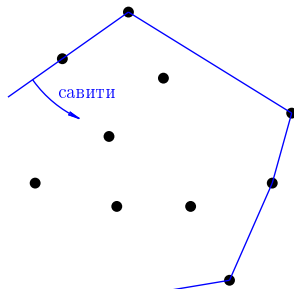
- Временска сложеност $O(n^2)$



Слика 4. Пример – корак 4

Алгоритам „паковање поклона” (gift wrap)

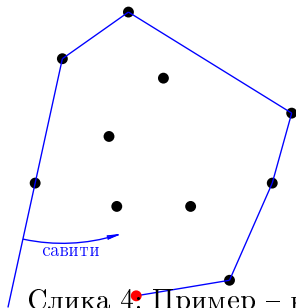
- Временска сложеност $O(n^2)$



Слика 4. Пример – корак 5

Алгоритам „паковање поклона” (gift wrap)

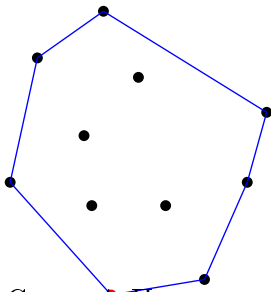
- Временска сложеност $O(n^2)$



Слика 4. Пример – корак 6

Алгоритам „паковање поклона” (gift wrap)

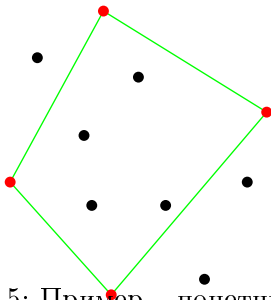
- Временска сложеност $O(n^2)$



Слика 4: Пример – омотач

„Брзи” алгоритам (quickhull)

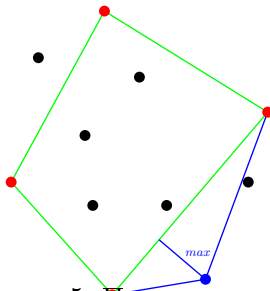
- Временска сложеност $O(n \log n)$



Слика 5: Пример – почетни четвороугао

„Брзи” алгоритам (quickhull)

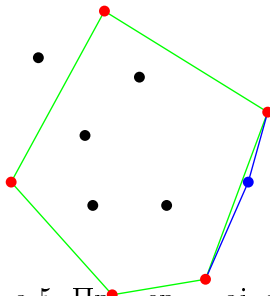
- Временска сложеност $O(n \log n)$



Слика 5: Пример – доњи десни

„Брзи” алгоритам (quickhull)

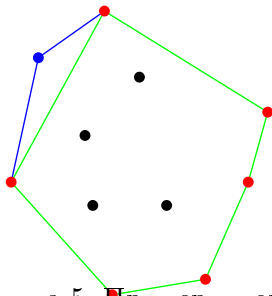
- Временска сложеност $O(n \log n)$



Слика 5: Пример – најдаља од нове

„Брзи” алгоритам (quickhull)

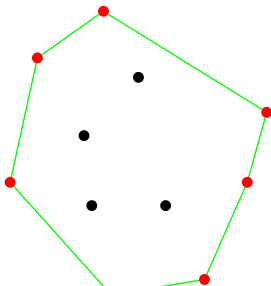
- Временска сложеност $O(n \log n)$



Слика 5: Пример – горњи и леви

„Брзи” алгоритам (quickhull)

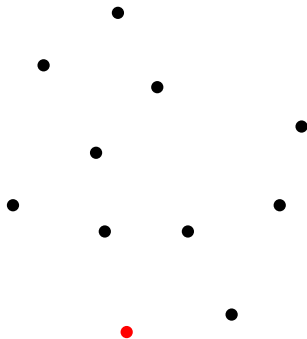
- Временска сложеност $O(n \log n)$



Слика 5: Пример – омотач

Грахамов алгоритам

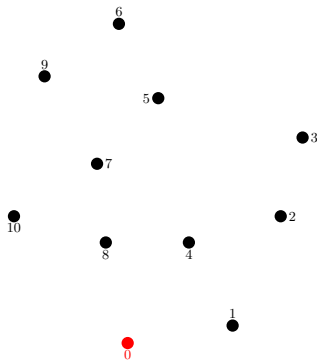
- Временска сложеност $O(n \log n)$



Слика 6: Пример – P_0 = најнижа (крајња десна) тачка

Грахамов алгоритам

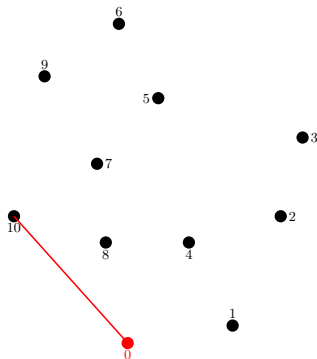
- Временска сложеност $O(n \log n)$



Слика 6: Пример – сортирање тачке (према углу)

Грахамов алгоритам

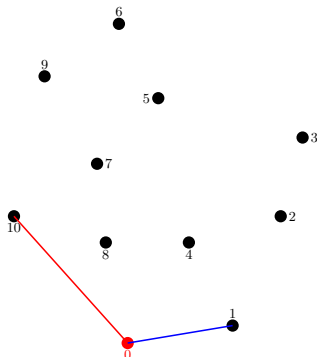
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: [10 0]

Грахамов алгоритам

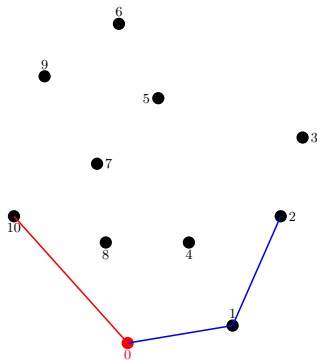
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: $[10 \ 0 \ 1]$

Грахамов алгоритам

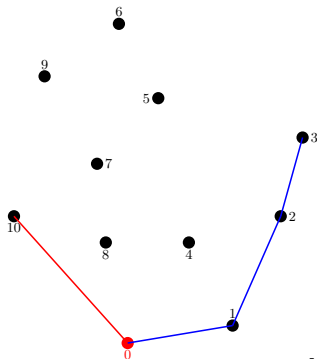
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: [10 0 1 2]

Грахамов алгоритам

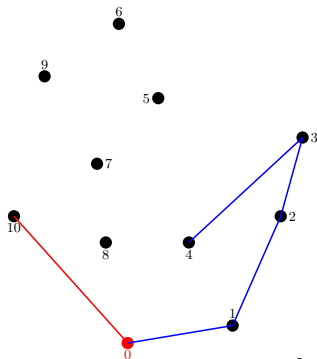
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: [10 0 1 2 3]

Грахамов алгоритам

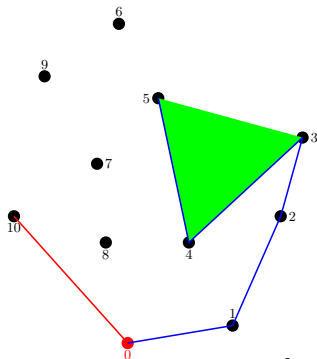
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: [10 0 1 2 3 4]

Грахамов алгоритам

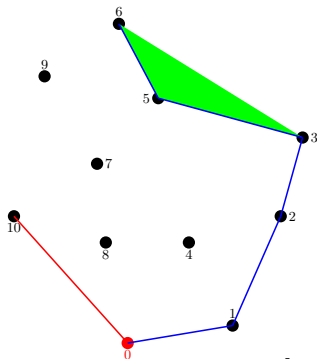
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: $[10\ 0\ 1\ 2\ 3\ \cancel{4}\ 5]$

Грахамов алгоритам

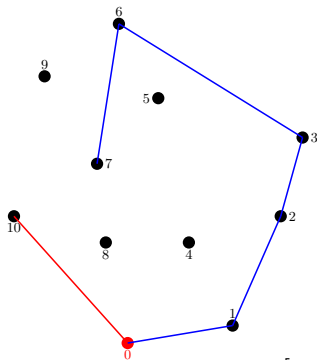
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: $[10\ 0\ 1\ 2\ 3\ \cancel{4}\ 6]$

Грахамов алгоритам

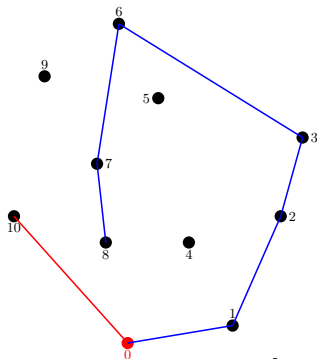
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: [10 0 1 2 3 6 7]

Грахамов алгоритам

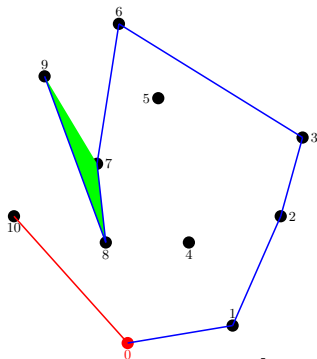
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: [10 0 1 2 3 6 7 8]

Грахамов алгоритам

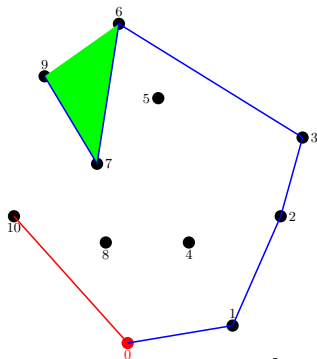
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: $[10 \ 0 \ 1 \ 2 \ 3 \ 6 \ 7 \ \cancel{8} \ 9]$

Грахамов алгоритам

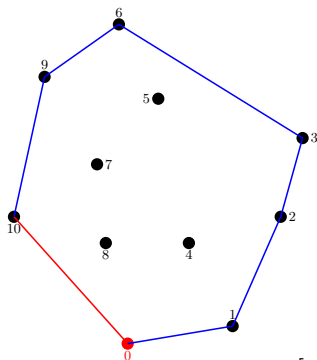
- Временска сложеност $O(n \log n)$



Слика 6: Пример – стек: $[10\ 0\ 1\ 2\ 3\ 6\ \cancel{7}\ 9]$

Грахамов алгоритам

- Временска сложеност $O(n \log n)$



Слика 6: Пример – омотач [10 0 1 2 3 6 9]

Примери

Пример 2

Одредити конвексни омотач скупа тачака $P_0 = (1, 3)$,
 $P_1 = (-2, 0)$, $P_2 = (-3, 5)$, $P_3 = (4, 2)$, $P_4 = (1, 1)$, $P_5 = (6, 4)$,
 $P_6 = (2, -3)$, $P_7 = (5, 5)$, $P_8 = (5, -1)$.

Задатак решити:

- а) Цртањем.
- б) Грахамовим алгоритмом.

Триангулација полигона

- Триангулација је разлагање неког лика на троуглове.

Триангулација полигона

- Триангулација је разлагање неког лика на троуглове.
- Триангулација простог полигона је разлагање његове унутрашњости унутрашњим дијагоналама које се међусобно не секу.

Триангулација полигона

- Триангулација је разлагање неког лика на троуглове.
- Триангулација простог полигона је разлагање његове унутрашњости унутрашњим дијагоналама које се међусобно не секу.

Лема 3.1

Сваки прост полигон са више од 3 темена има унутрашњу дијагоналу.

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ▶ ↺ 🔍 ↻

Примери

Пример 3

Од датих тачака у равни формирати прост полигон, а затим га триангулисати.

- а) $P_0 = (0, 0)$, $P_1 = (5, -1)$, $P_2 = (3, 2)$, $P_3 = (6, 4)$,
 $P_4 = (-1, 3)$.
- б) $P_0 = (-1, 3)$, $P_1 = (2, 1)$, $P_2 = (0, 0)$, $P_3 = (4, -1)$,
 $P_4 = (5, 3)$, $P_5 = (3, 4)$.

Проблем уметничке галерије

Проблем: Поставити минималан број чувара који покривају читаву галерију.

Проблем уметничке галерије

Проблем: Поставити минималан број чувара који покривају читаву галерију.

Галерија = прост полигон са n ивица;

Чувари = тачке унутар полигона.

Проблем уметничке галерије

Проблем: Поставити минималан број чувара који покривају читаву галерију.

Галерија = прост полигон са n ивица;

Чувари = тачке унутар полигона.

Chvátal: Горња граница = $\frac{n}{3}$ чувара!

Проблем уметничке галерије

Проблем: Поставити минималан број чувара који покривају читаву галерију.

Галерија = прост полигон са n ивица;

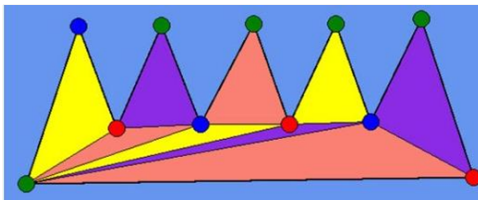
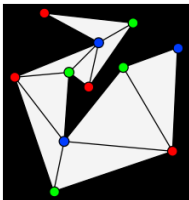
Чувари = тачке унутар полигона.

Chvátal: Горња граница = $\frac{n}{3}$ чувара!

Алгоритам:

- Триангулисати полигон;
- Обојити темена подеоних троуглова (3-бојење);
- Изабрати за чуваре темена обојена истом бојом.

Проблем уметничке галерије – примери



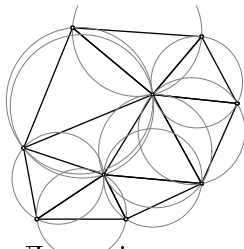
Слика: Примери: Чувари – плаве тачке

Алгоритми за триангулацију полигона

- унутрашњим дијагоналама
- „завртањем ушију”
- триангулација монотоних полигона и монотоних планина
- Делонијева триангулација
- триангулација у линеарном времену

Слика 8: Делонијева триангулација

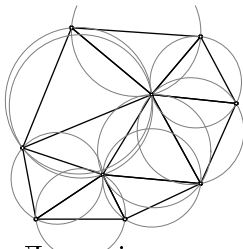
Делонијева триангулација



Слика 8: Делонијева триангулација

Делонијева триангулација **минимизује максималан полупречник круга** описаног око троугла триангулације.

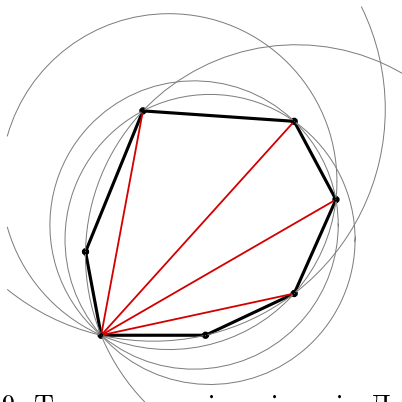
Делонијева триангулација



Слика 8: Делонијева триангулација

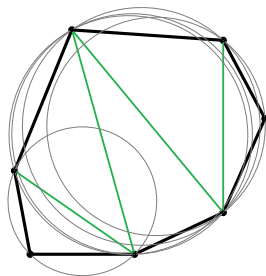
Делонијева триангулација **минимизује** **максималан** **полупречник круга** описаног око троугла триангулације. Поступком максимизације најмањег угла се **не минимизује** највећи угао, **нити се минимизују** дужине страница.

Делонијева триангулација - примери



Слика 9: Триангулација која није Делонијева

Делонијева триангулација - примери

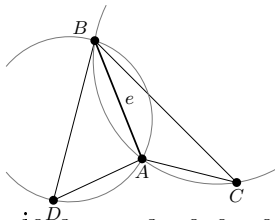


Слика 9: Делонијева триангулација

Локално Делонијеве ивице

Дефиниција 3.1

Унутрашња ивица AB је **локално Делонијева** ако тачка D не припада унутрашњости круга који садржи тачке A , B и C .

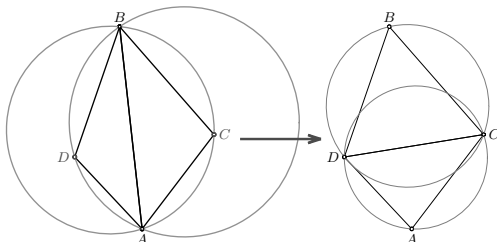


Слика 10: Делонијева ивица неконвексног четвороугла

Теорема 3.2

Ако је \mathcal{T} триангулација чије су све ивице локално Делонијеве, тада је \mathcal{T} Делонијева триангулација.

Обртање ивица (flip algorithm)



Слика 11: „Обртање ивице”

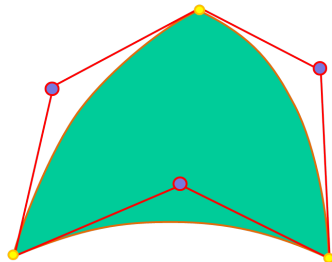
Теорема 3.3

Алгоритам „обртања ивица” се зауставља након

$\binom{n}{2} = O(n^2)$ корака и његов резултат је Делонијева

триангулација која максимизује најмањи угао на скупу свих триангулација.

„Криволинијска” триангулација¹



Слика: Безијев троугао са контролним полигоном

- Пример 1
- Пример 2
- Пример 3

¹Материјале уступио Hagen Wille