ОБУЧЕНИЕ ЧРЕЗ ЗАПИСВАНЕ НА СЛУЧАИ

Лекция 13

<u>Въведение.</u>

- **Сега ще се занимаем с това,
  как бихме могли да се справим с обученито в области,
  в които добри модели е невъможно да се построят.**

- **В такива случаи бихме могли да се обучаваме
  като записваме конкретни примери без никаква обработка
  на съдържаната в тях информация.**

- **Ще се запознаем как да използваме
  евристиката за <u>устойчивост на свойствата</u>
  на наблюдаваните вече обекти
  при определянето свойствата на невиждани до сега обекти
  и как да намерим най-близкия им съсед в пространството на свойствата.**

- **Накрая ще се научим как да намираме най-близкия съсед
  в пространството на свойствата**

Евристиката устойчивост на свойствата

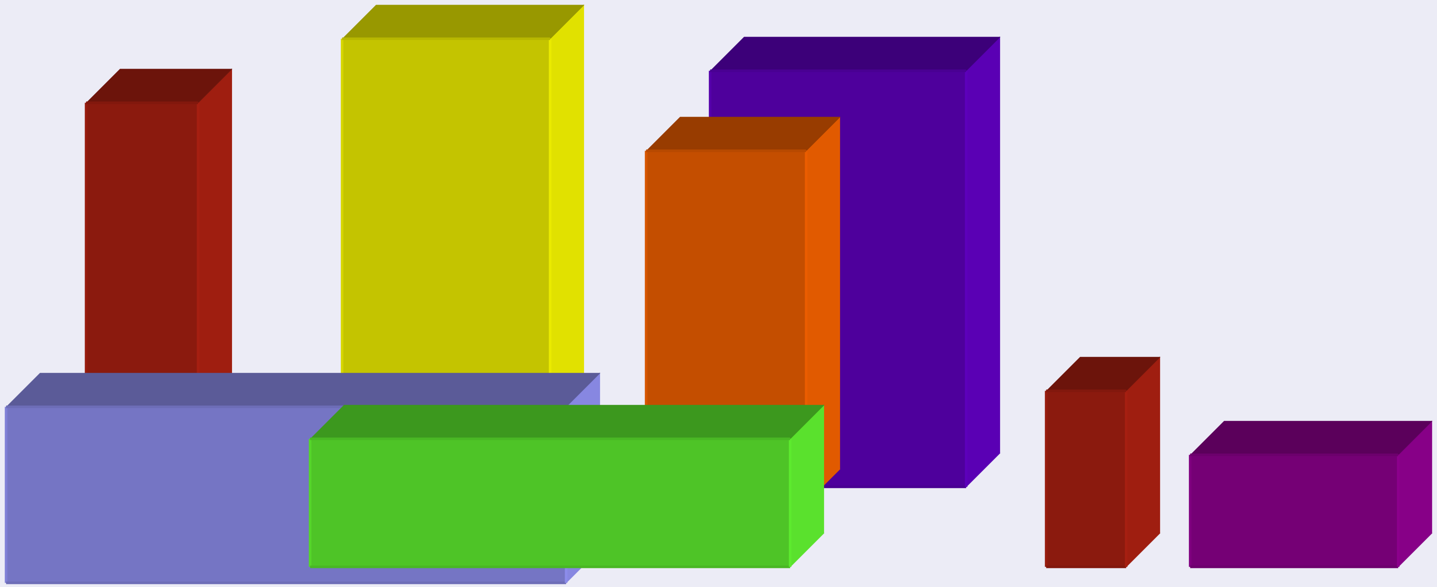
Да започнем с евристиката устойчивост на свойствата
и разпознаването на типа проблеми,
които могат могат да се решават чрез записването на
наблюдавани случаи с цел те да се използват при необходимост.

Евристиката устойчивост на свойствата прави възможно
да си спомним за наблюдавани вече случаи по налични техни свойства.

Евристиката устойчивост на свойствата

Да започнем с пример.
Зададени са осем блокчета.
За всяко блокче са ни известни три характеристики:
цвят, височина и ширина.
Нека добавим към тези блокчета ново блокче
с височина 4 см. и ширина 1 см.
Какъв цвят трябва да има новото блокче?

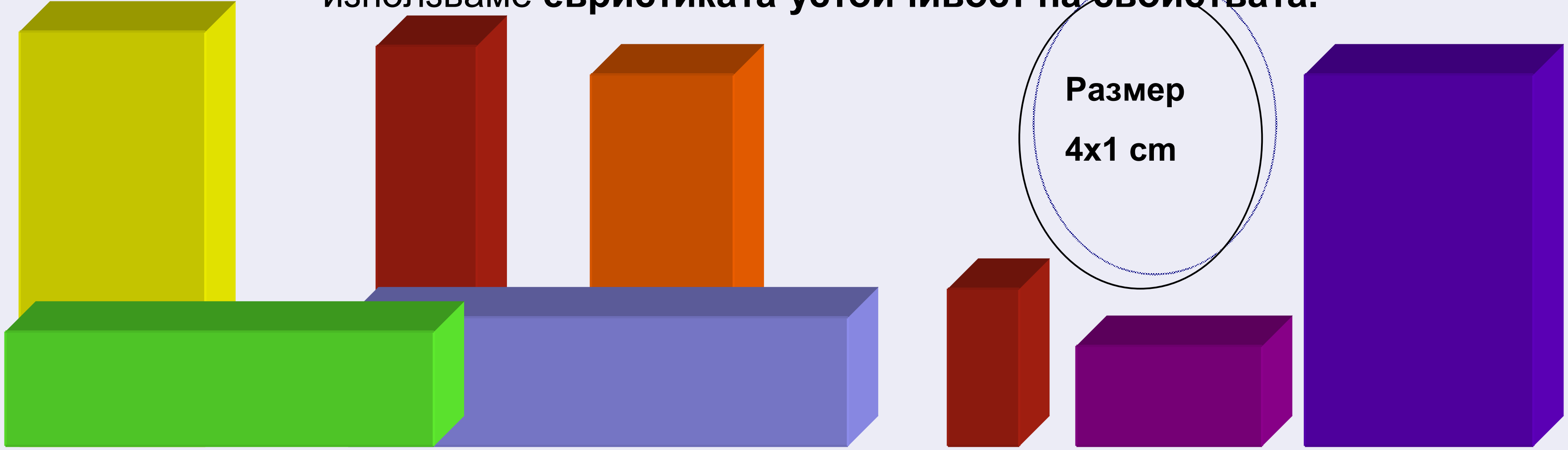
Евристиката устойчивост на свойствата



Евристиката устойчивост на свойствата

Ако не разполагаме с повече сведения,
здравият разум ни съветва да предположим,
че цветът на новото блокче е същия,
като този на блокчето,
чиито други характеристики са най-близки до неговите.
В случая бихме предположили,
че това блокче има същия цвят
както блокчето с най-близки ширина и височина.

За да направим това предположение,
използваме **евристиката устойчивост на свойствата.**

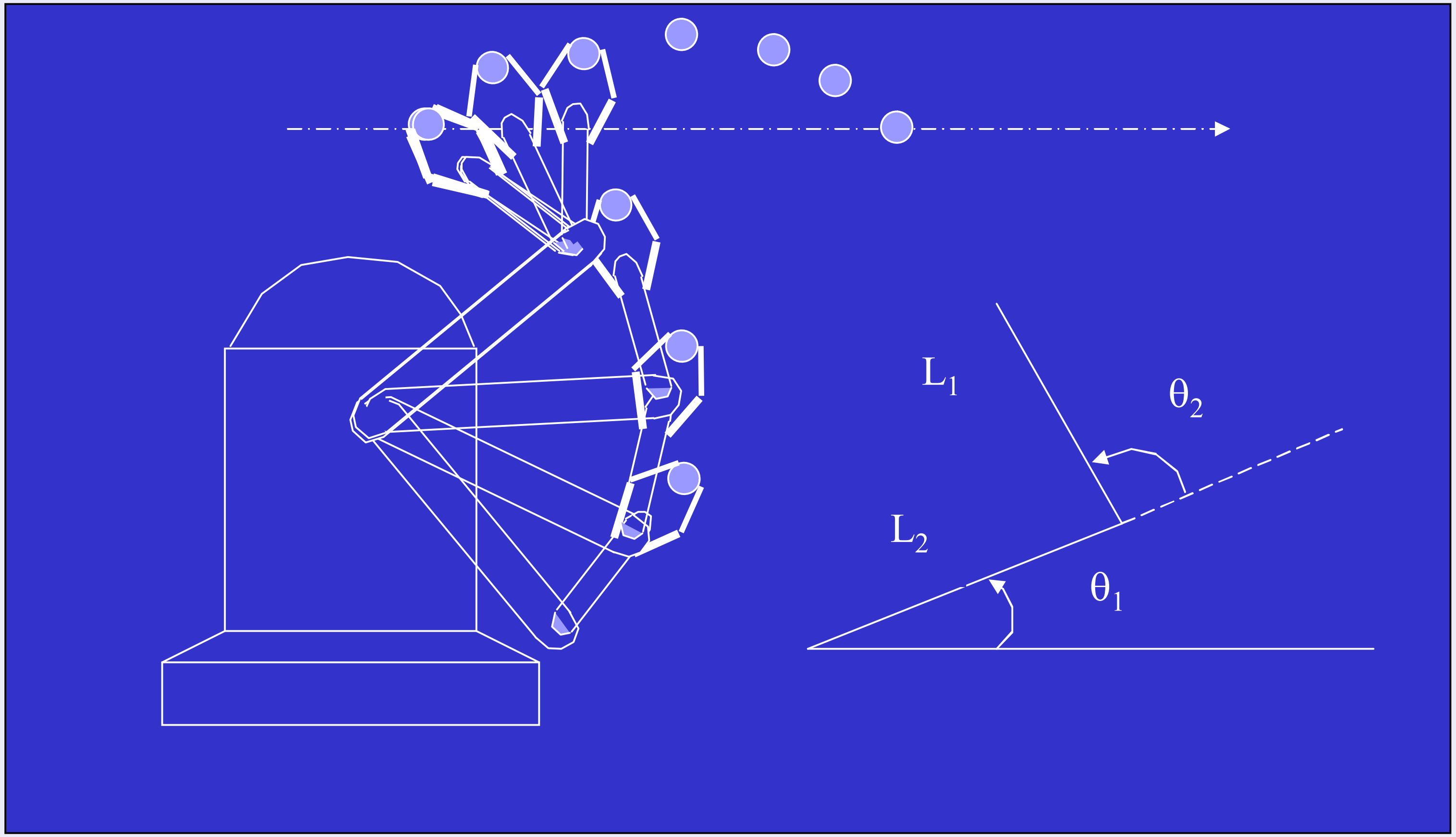


Използуване на евристиката устойчивост на своиствата.

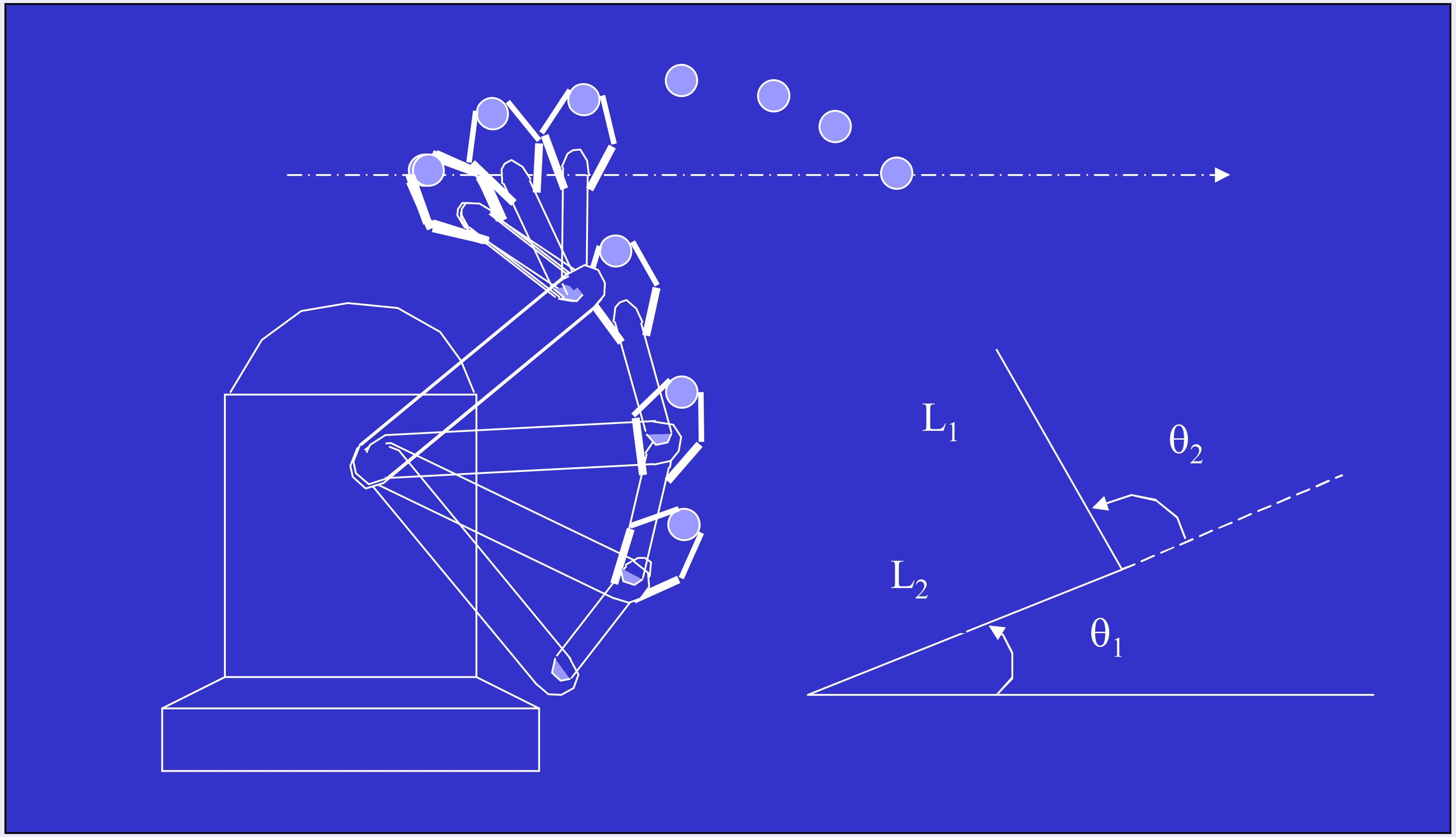
(1)Когато е необходимо да се намери
стойност на свойство на обект или ситуация,
без да е дадена друга информация освен множество от препоръчителни случаи,
тогава от множеството препоръчителни случаи намери чрез познатите свойства на новия обект най-подобния му случай.

(2)Предположи,
че непознатото свойство на новия обект е същото,
както свойството на намерения най-подобен му обект.

Разглеждаме движение на ръката на робот
при зададен път в пространството и при зададена скорост.
Необходимо е да се знае
как се променя ъгълът на ставата,
какъв ъгъл на усукване ще е необходим и т.н.
Моделирането на пътя е сложен кинематичен проблем.
Формулите за управление на манипулатора
– позициониране и ориентиране на ставите - са много обемни и изискват сложни изчисления.



Нещата стават на порядък по-сложни,
ако се развиват в тримерно пространство
и се вземат под внимание външни влияния,
като земното привличане и т.н., подробности,
без които движението на ръката не би било плавно и точно.



Има и друг подход към решаване на такива задачи.

Съставя се таблица с колони за
въртящия момент, позициите, скоростите, квадрата на скоростите,
производните на скоростите и ускоренията,
с други думи за всички необходими параметри на движението.

Когато е необходимо да се премести ръката на робота по дадена траектория,
се използва тази таблица като пространство с характеристики
и в нея се намират близки до необходимите характеристики за всяка позиция.

Позициите могат да се интерполират,
за да се намери подходящо задвижване,
съответстващо на всяка малка отсечка от траекторията, която ни интересува.

Има и друг подход към решаване на такива задачи.

Единственото изискване е,
че таблицата трябва да съдържа достатъчно записи,
за да попълни пространството на характеристиките достатъчно гъсто.

След няколко опита
движенията стават плавни и точни,
защото вече се интерполира измежду значително по-близко разположени точки
от траекторията на ръката на робота.

Евристиката устойчивост на свойствата
помага и при решаването на сложни динамични проблеми

Решаващо дърво и неговото построяване.

Един начин да се открие най-близкия съсед на дадено блокче е да се изчисли разстоянието от това блокче до всяко друго блокче и после да се намери минималното измежду тези разстояния.

За **n** различни блокчета разстоянията са на брой **n-**1.
Този подход е добър,
ако броят на блокчетата е малък (например 10),
но никак не е добър,
ако броят на блокчетата е голям (милион и повече).

Решаващо дърво и неговото построяване.

Има по-добър начин за решаването на тази задача,
при който броят на изчисленията е пропорционален на $\log_2 n$.
Този подход използува специален вид решаващо дърво.

Решаващото дърво се състои от така подредени тестове,
че при изпълнението си
всеки тест посочва най-подходящия
следващ тест за изпълнение.

Решаващото дърво е семантично дърво

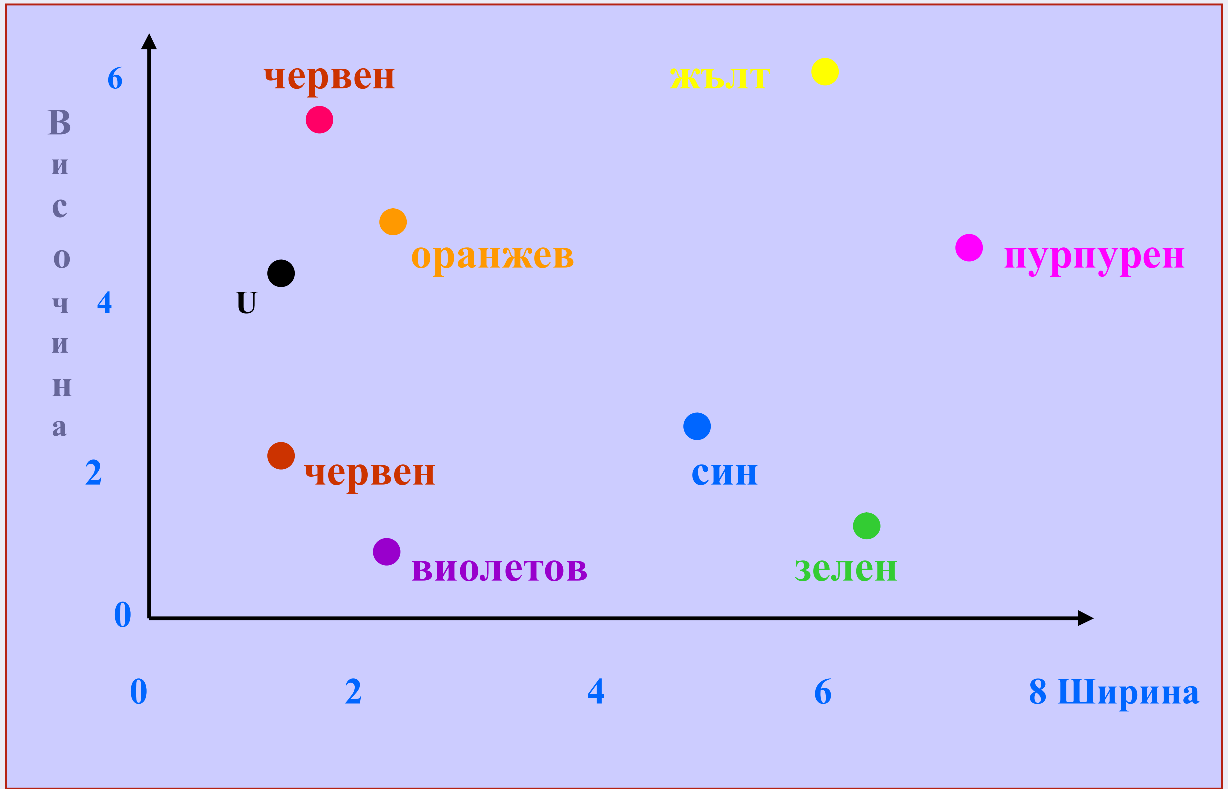
<u>В което:</u>

Всеки връх е свързан с множество от възможни отговори.

Всеки връх, който не е листо,
е белязан с тест,
чиито възможни отговори
разделят множеството обекти на подмножества.

Всяка дъга в разклонението
пренася съответното си подмножество
към следващ наследяващ връх.

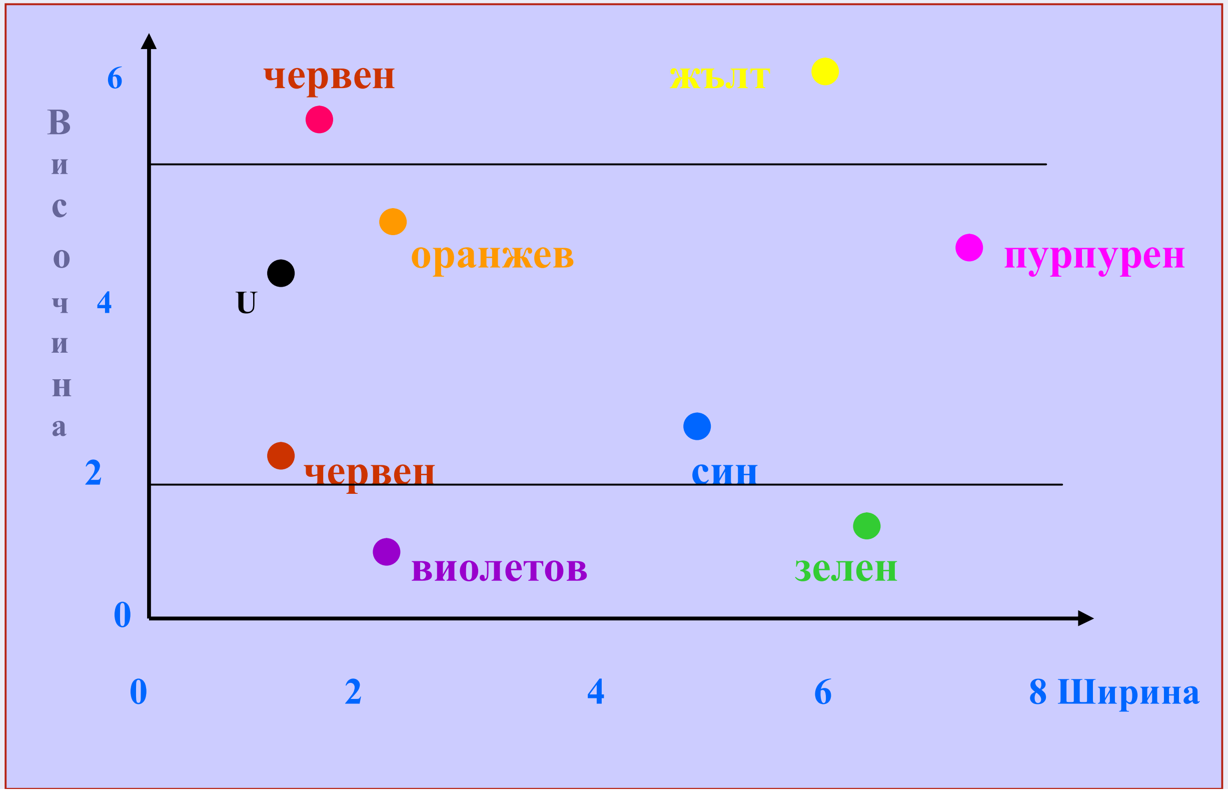
На фигурата е показано разположението на блокчета
в т.н. пространство на характериките,
което за нашия случай е с две координати – ширина и височина,
тъй като това са известните ни характеристики за всички блокчета
в това число и онова с неизвестния цвят.
Връзката на блокчето с неизвестен цвят с осемте блокчета с познат цвят
може да се интерпретира като търсене на най-близкия съсед
в това пространство от известни характеристики



Как се строи решаващо дърво.

- **<u>Първа стъпка.</u>**

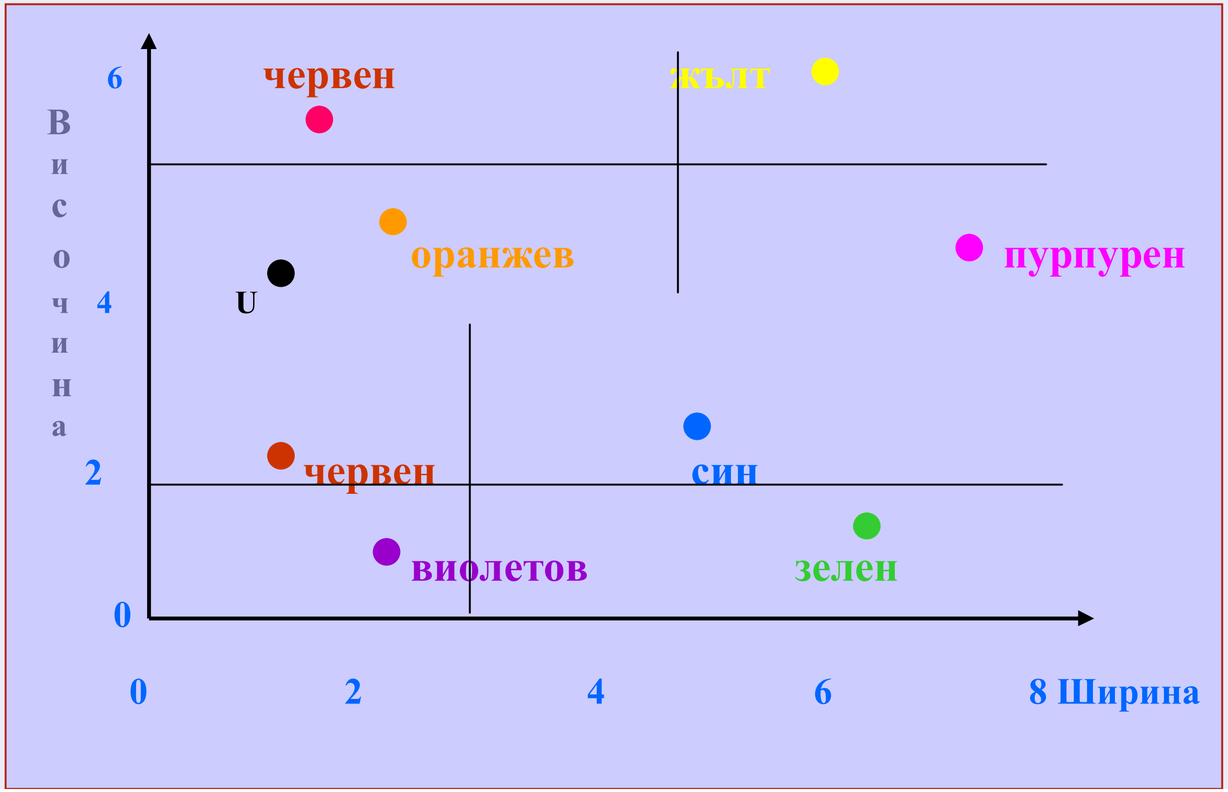
Блоковете с известен цвят се разделят
според височината им
на подмножества.
Високите блокове се разделят на блокове са с височина по-голяма от 5см., и блокове по-ниски от 5см
Ниските блокове се разделят на блокове са с височина по-голяма от 2см., и блокове по-ниски от 2см.



Как се строи решаващо дърво.

- **<u>Втора стъпка.</u>**

Подмножеството на високите блокове се разделя на подмножество,
в което всички ширини са равни или по-големи от 5см.
и друго подмножество, в което всички ширини са равни или по-малки от 3см.
По подобен начин подмножеството ниски блокове се разделя на
подмножество от блокове с ширина по-малка или равна на 2см.
и с ширина по-голяма или равна на 4см.

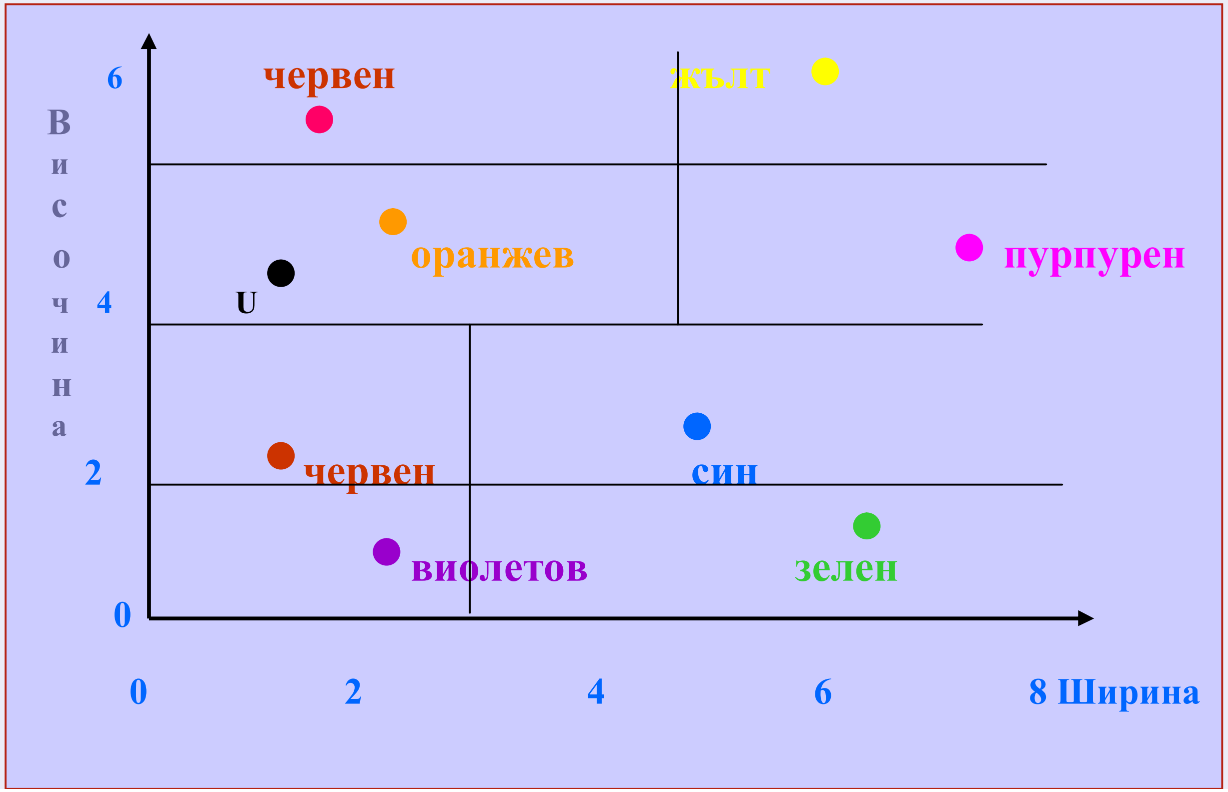


Как се строи решаващо дърво.

- **Трета стъпка.**

Следва разделяне по **височина** така,че да се получат осем подмножества от по един блок всяко.

Резултатът се нарича **k-d дърво**, където **к е размерността на** пространството, т.е. броят на използваните свойства, а **d е дълбочината на дървото.** За случая дървото е 2-3 дърво.

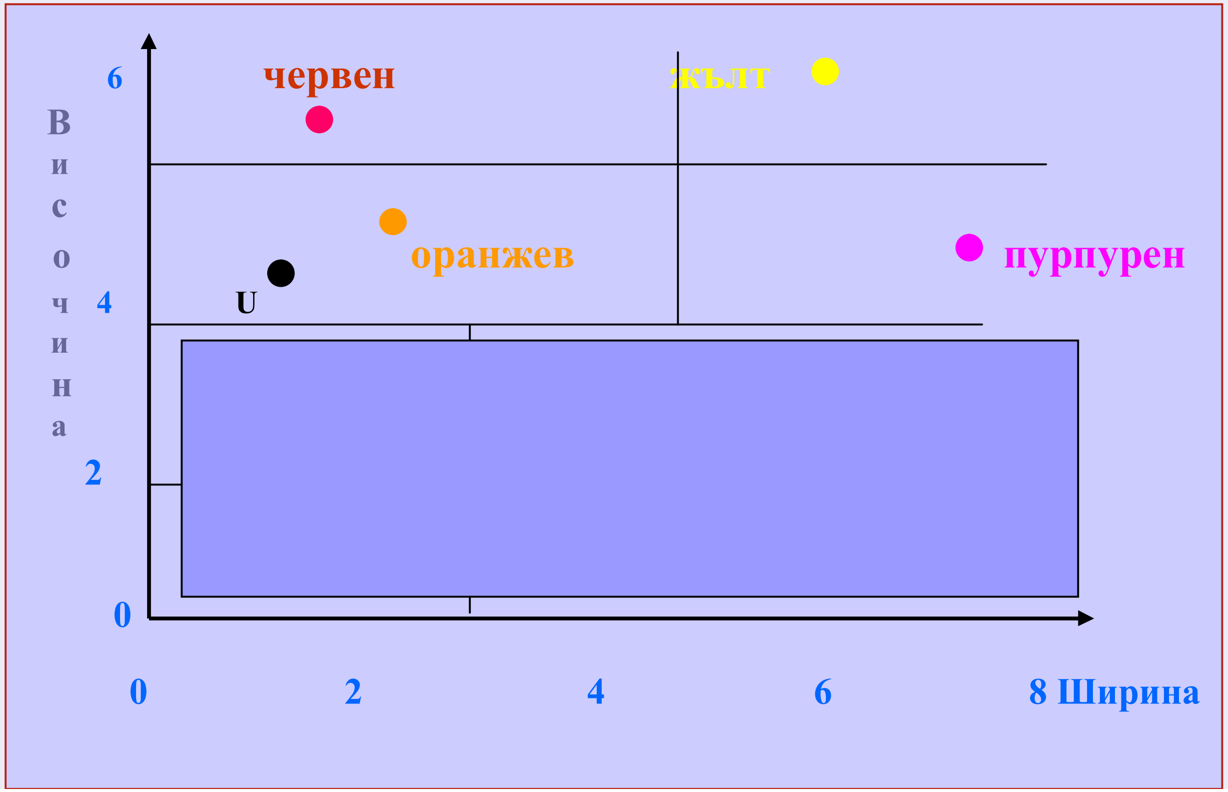


- **Намиране на най-близкия съсед на блока с неизвестен цвят**

- Първо

Тъй като *височината* на **U** е повече от **3.5см**.

- ✓ Това е между височината на най-ниския от високите блокове
- ✓ и най-високия от ниските блокове.
- ✓ Решаваме, че **U** е по-близо до **високите блокове**
- ✓ и игнорираме разглеждането на ниските блокове.



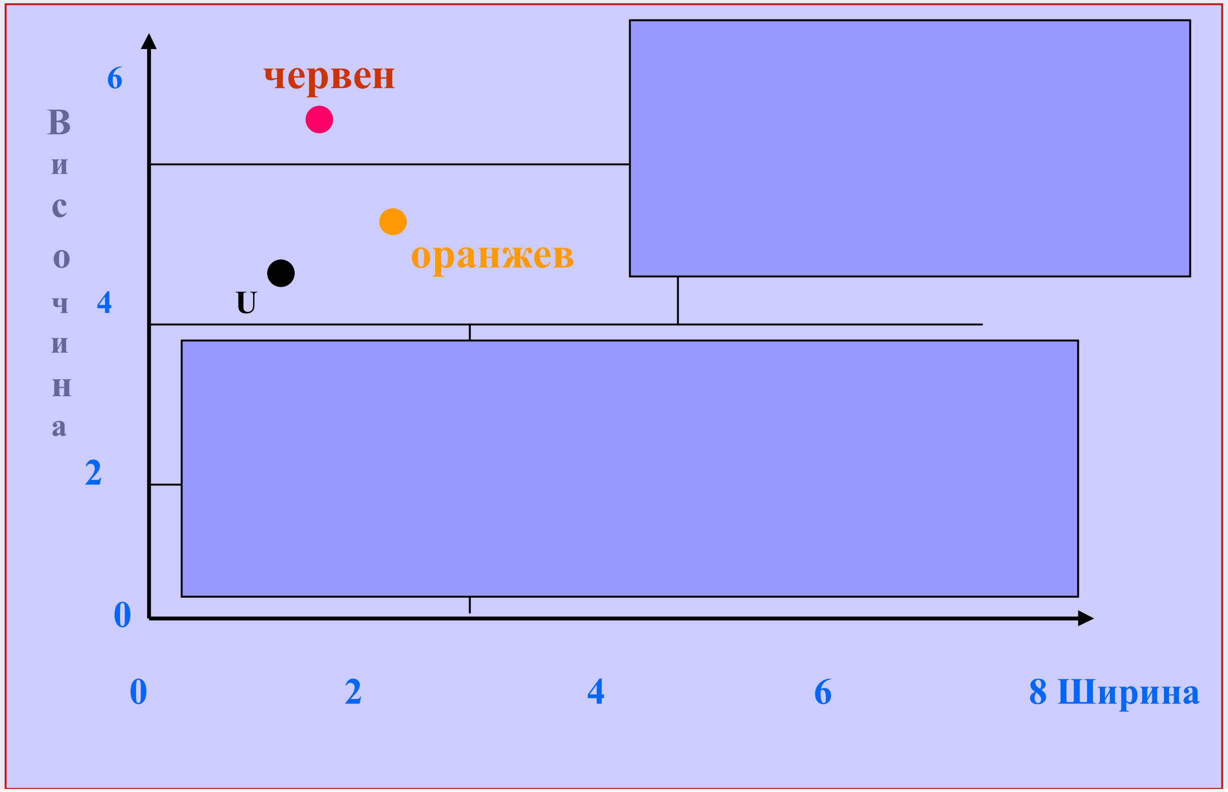
Намиране на най-близкия съсед на блока с неизвестен цвят

- След това

тъй като *ширината* на **U** е по-малка от **3.5см**.

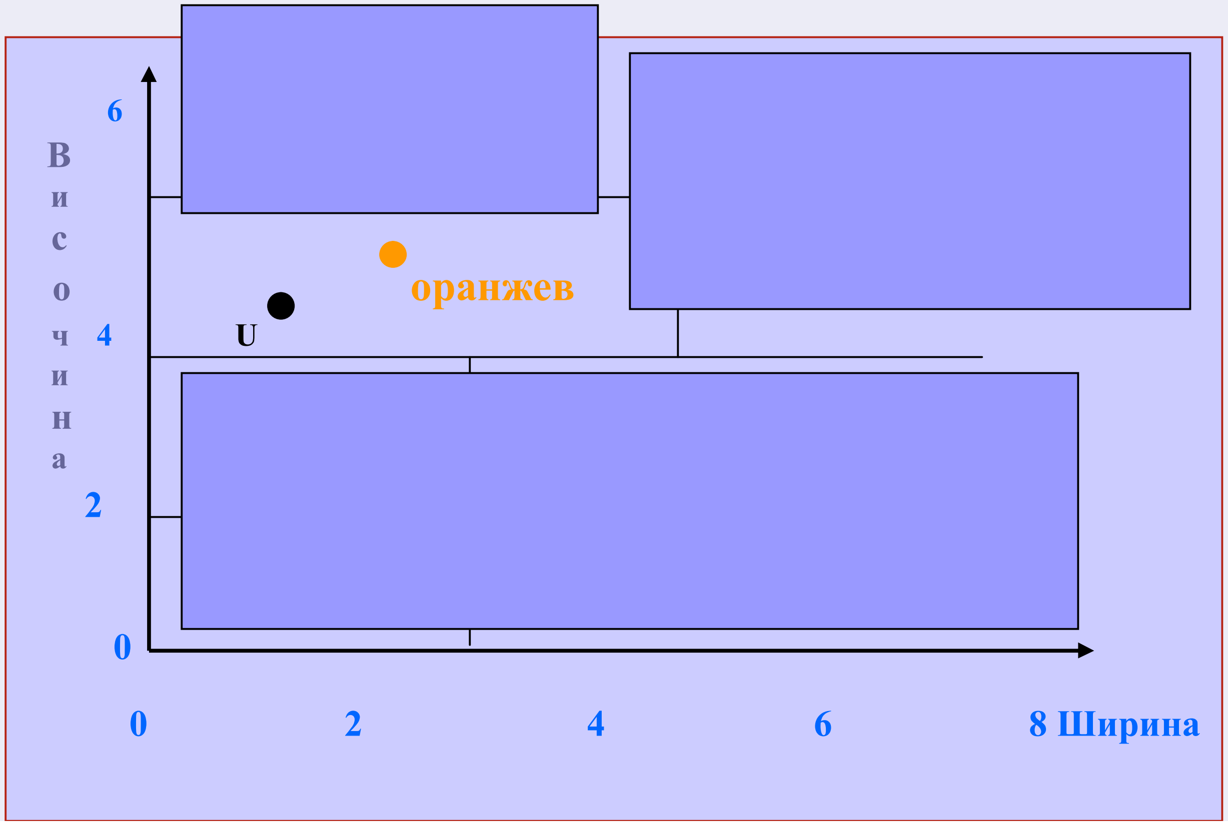
✓ по-вероятно е търсеният съсед да е по-близо до тесните блокове

✓ и игнорираме разглеждането на широките блокове.



- **Намиране на най-близкия съсед на блока с неизвестен цвят**

- И накрая,
- тъй като височината на U е по-малка от 5,5см,
- ✓ попадаме при оранжевото блокче.
- ✓ Тогава приемаме,
- ✓ че най-близкият съсед на блокчето U е оранжевото блокче.



Намиране на най-близкия съсед на блок U.

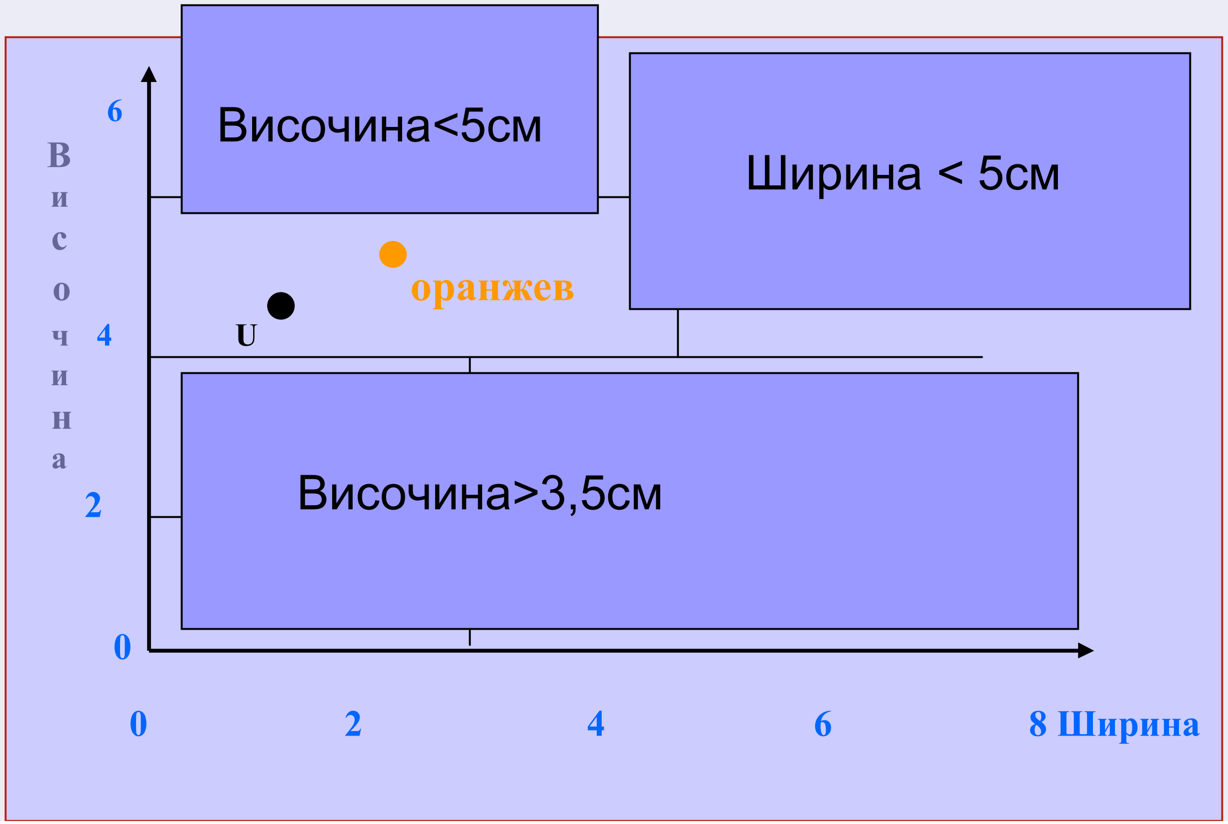
Така намирането на най-близкия блок е
следване на път през решаващото дърво.
За нашия пример бяха достатъчни три сравнения.

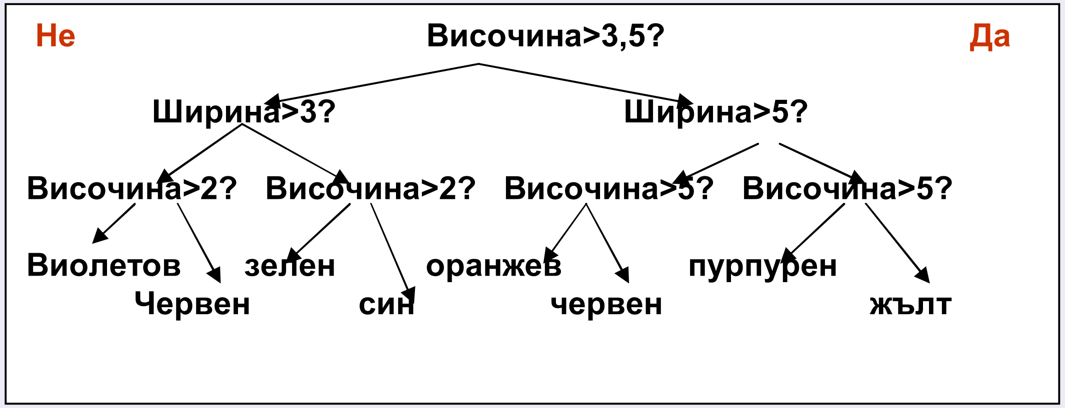
Решаващо дърво с фактор на разклонение 2 и дълбочина d
ще има **2^d разлиствания**.
Ако **обектите са n** за дълбочината на дървото **d** се получава зависимостта
$2^d \geq n$.

За необходимия брой сравнения,
който е равен на дълбочината на дървото,
се получава зависимостта **$d \geq \log_2 n$.**

Ако за 8 обекта ползата не е голяма,
то за 1 милиард обекта сравненията са от порядъка на 30.

- Намиране на най-близкия съсед на блока с неизвестен цвят.





k-d дървото е решаващо дърво,
в което:

- Всяко листо съдържа по един обект;
- Всеки тест посочва координата (праг, около който няма обекти);
- Всеки тест разделя множеството от обекти на две подмножества.

Нека приведем пълната процедура за разделянето на подмножества и едвременното построяване на решаващото дърво.

<u>За да се разделят случате на подмножества</u>:

- **1) Ако случаят е единствен следва стоп.**
- **2) Ако това е първото разделяне на случаите, избери ос за разделяне,**

в противен случай избери ос за разделяне, която е различна от избраната ос в предишното разделяне.

- **3) Намери двата най-близки до разделящата подмножествата ос обекта съответно от едното и от другото подмножества.**

Постави разделящата ос по средата между тези два обекта и наречи координатата й праг.

Построй тест в решаващото дърво с този праг.

Като имате предвид само остта за разделяне намерете средна позиция на два средни за подмножествата обекта.

Наречи тази средна позиция праг и построй тест в решаващото дърво, който сравнява неизвестния обект в осите на разделяне с прага.

Отбележи позициите на двата най-близки до разделящата подмножествата ос обекта върху оста на разделяне.

Наречи тези позиции горна и долна граница.

- **4) Раздели всички обекти на две подмножества в съответствие с това от коя страна на разделящата ос се намират.**
- **5) Раздели обектите на всяко подмножество,**

като формираш поддърво за всяко подмножество, следвайки същата процедура.

Процедура за намиране на най-близкия съсед

Процедура k-d за намиране на най-близкия съсед:

1)Определя се дали има само един елемент в множеството, което разглеждаме. Ако това е така, това е най-близкия съсед.

2)В противен случай неизвестният обект се сравнява с прага на текущия връх от дървото. Резултатът от това сравнение води към подмножество, което вероятно съдържа най-близкия съсед.

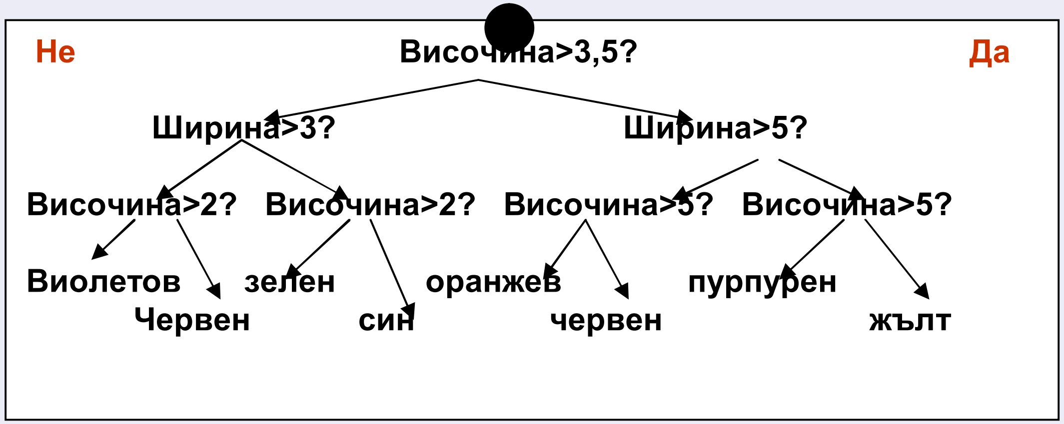
3)Намира се най-близкия съсед в това подмножество като се използва тази процедура.

4)Определя се дали разстоянието до най-близкия съсед в това подмножество е по-малко или равно на разстоянието до граничните линии (горна/долна граници) от осите за разделяне.

а)Ако това е така, най-близкият съсед е намерен.

б)Ако това не е така, провери другото подмножество като използваш тази процедура.

Намирането на най-близкия съсед



Блокът с неизвестен цвят има височина 4 см. и ширина 1 см.

Намирането на най-близкия блок
се свежда до следване на път през решаващото дърво

Заключение.

1)Евристиката за устойчивост на свойствата
е основание за използването запомнени случаи
като източник на сведения за свойства за обекти,
които не сме срещали преди.

2)Запомнените обекти могат да ни помогнат
при решаването на много трудни проблеми,
включително и проблеми с динамичното управлението.

3)K_D процедурата е бърза последователна процедура,
която намира най-близкия съсед за логаритмично време.

Литература.

[Friedman, J.H., J.L.Bentley and Raphael, An Algorithm for Finding Best Matches in Logarithmic Time, ACM Transactions of MathematicalSoftwware, vol 3, no 3, 1977]

– Въвеждат се k-d дървета и се описва тяхното използуване.

[Atkeson,C.G., Memory-Based Approacess to Approximating Continuous Functions, Proceedings of the Workshop on Nonlinear Modelingand Forecasting, 1990.]

-Описва се паралелен алгоритъм, при който едновременно се обработват всички случаи.