

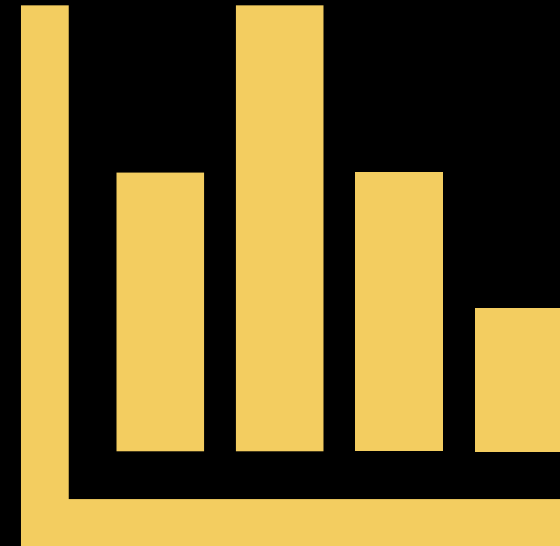
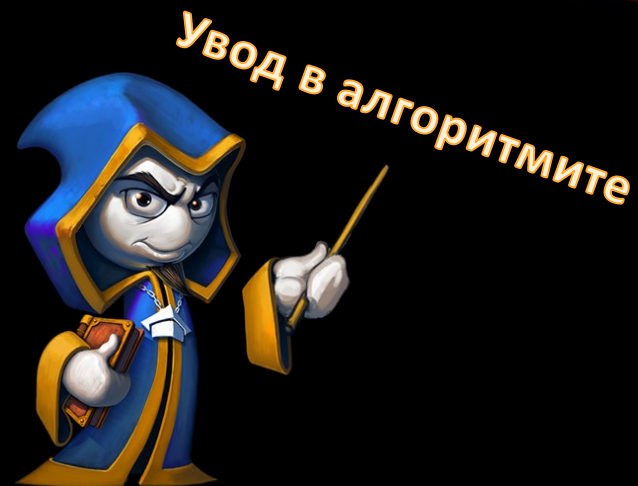
Въведение в сортирането, сортиране чрез пряка селекция



Учителски екип

Обучение за ИТ кариера

<https://it-kariera.mon.bg/e-learning/>



Съдържание

1. Какво е това сортиране?
2. Класификация на методите за сортиране
3. Стабилност на сортирането
4. Сортиране чрез пряка селекция



Що е това алгоритъм за сортиране?

- Сортиращ алгоритъм

- Алгоритъм, който подрежда елементите на списък

- В ненамаляващ ред

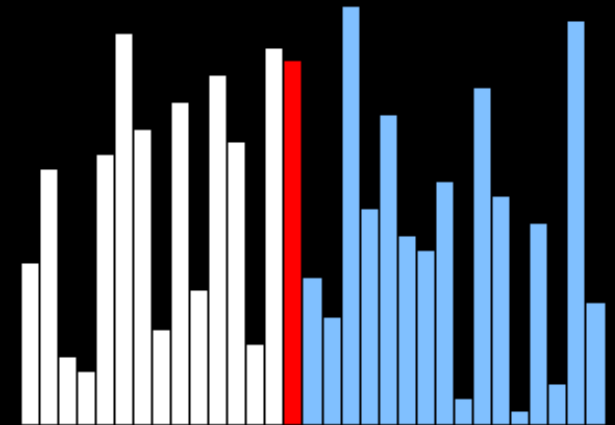
- Елементите трябва да са **сравними**

- По-формално обяснение

- **Входът** е последователност / списък от елементи

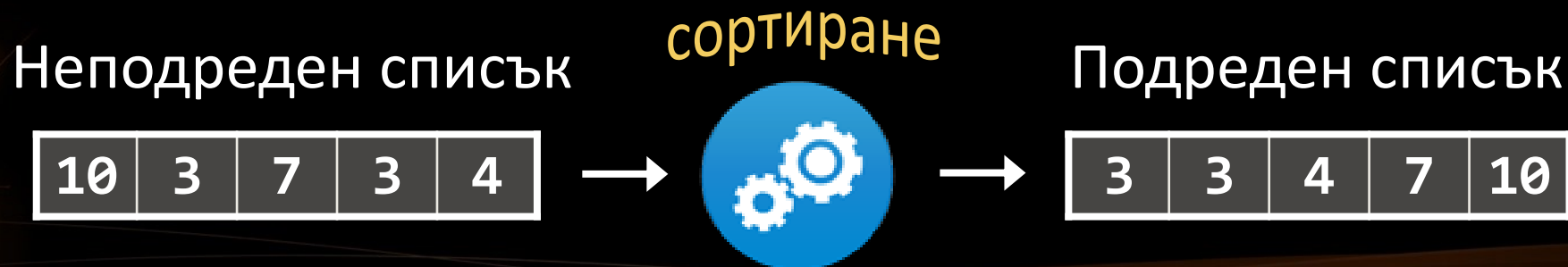
- **Изходът** е пренаареждане / **пермутация** на елементи

- В ненамаляващ ред



Пример за сортиране

- Ефективните алгоритми за сортиране са важни за
 - Генериране на разбираеми за човека изходни данни
 - Канонизирани данни – подsigуряване, че данните са подредени по специфичен начин
 - В комбинация с други алгоритми, например двоично търсене
- Пример за сортиране:

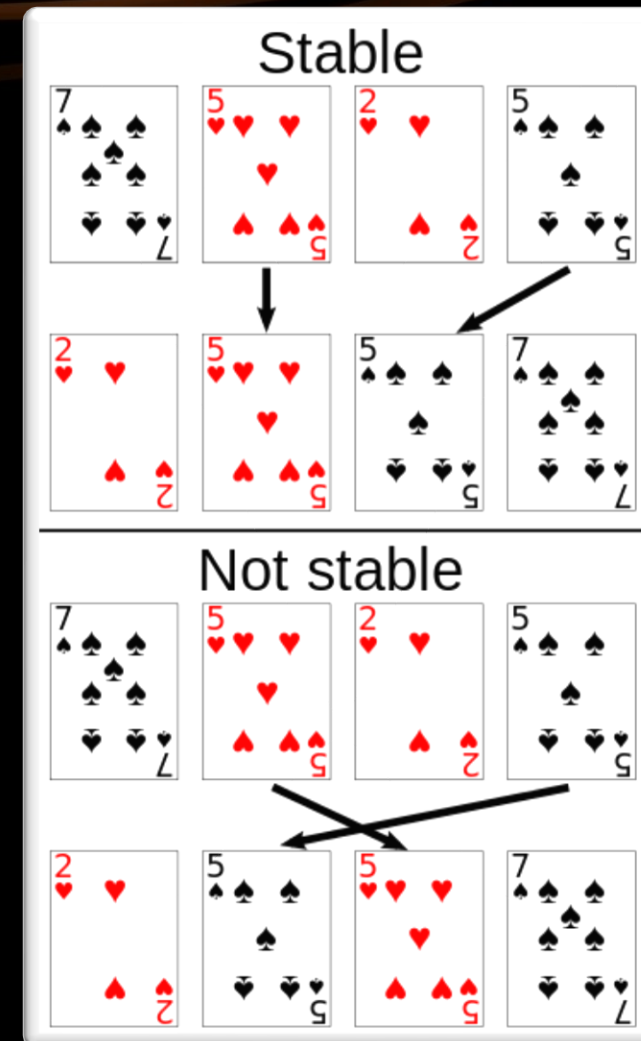


Класификация на алгоритмите за сортиране

- Сортиращите алгоритми често са класифицирани според
 - Изчислителната **сложност** и обема на използваната памет
 - Как се държат в най-лошия, обичаен и най-добрия случай
 - **Рекурсивни** / **нерекурсивни**
 - **Стабилност** – **стабилни** / **нестабилни**
 - **Базирано на сравнение** сортиране / **неизползващи сравнение**
 - **Метод** за сортиране: **вмъкване**, **замяна** (метод на мехурчето и бързо сортиране), **селекция** (пирамидално сортиране), **сливане**, **последователно** / **паралелно** и т.н.

Стабилност на сортирането

- **Стабилни** сортиращи алгоритми
 - Запазват подредбата на еднаквите елементи
 - Ако два елемента след сравнение са равни, редът им един спрямо друг се запазва
- **Нестабилни** сортиращи алгоритми
 - Пренареждане на еднаквите елементи в непредсказуем ред
- Често **различни елементи** имат **един и същ ключ**, използван за сравнение при сортиране



Помощни методи при сортиране

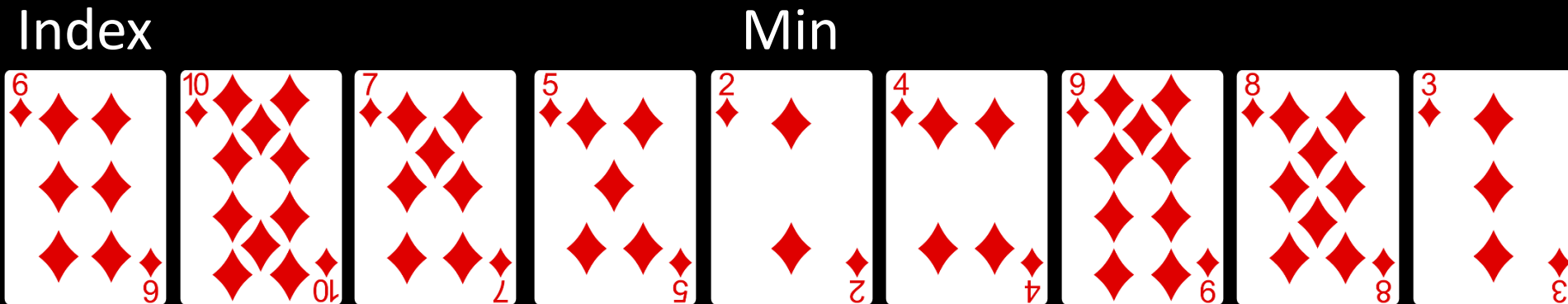
```
static void Swap<T>(T[] collection, int from, int to)
{
    //TODO: Размяна на елементите
}
```

```
static bool Less(IComparable first, IComparable second)
{
    return first.CompareTo(second) < 0;
}
```

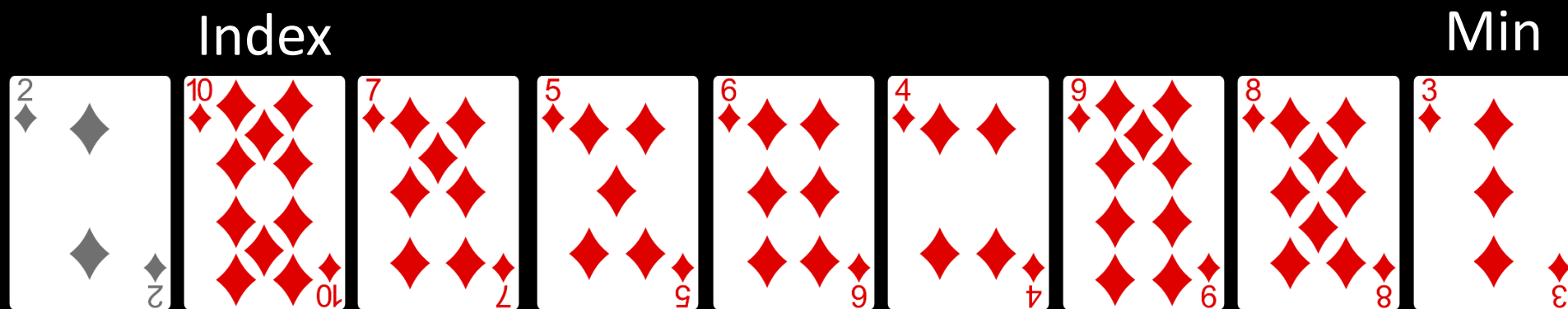
Сортиране чрез пряка селекция

- Сортиране чрез пряка селекция (selection sort) – прост, но неефективен алгоритъм (онагледяване)
 - Размяна на първия с минималния от елементите отдясно, после втория, третия и т.н.
 - Памет: $O(1)$
 - Стабилност: Не
 - Метод: Селекция

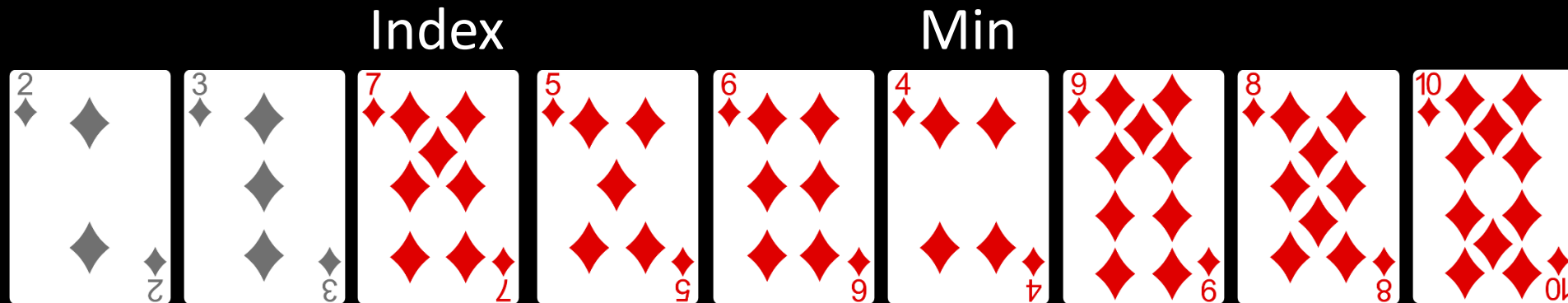
Пример за сортиране чрез пряка селекция



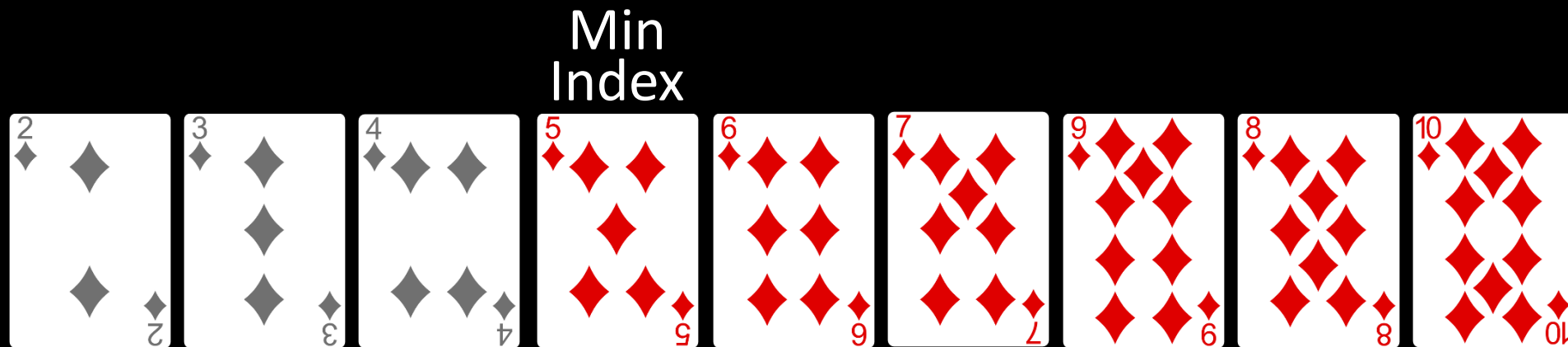
Пример за сортиране чрез пряка селекция



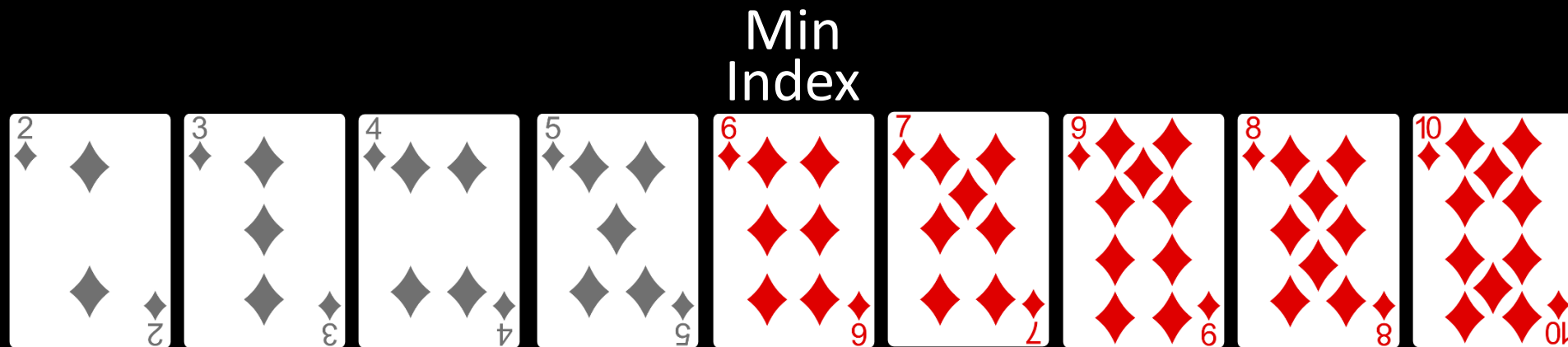
Пример за сортиране чрез пряка селекция



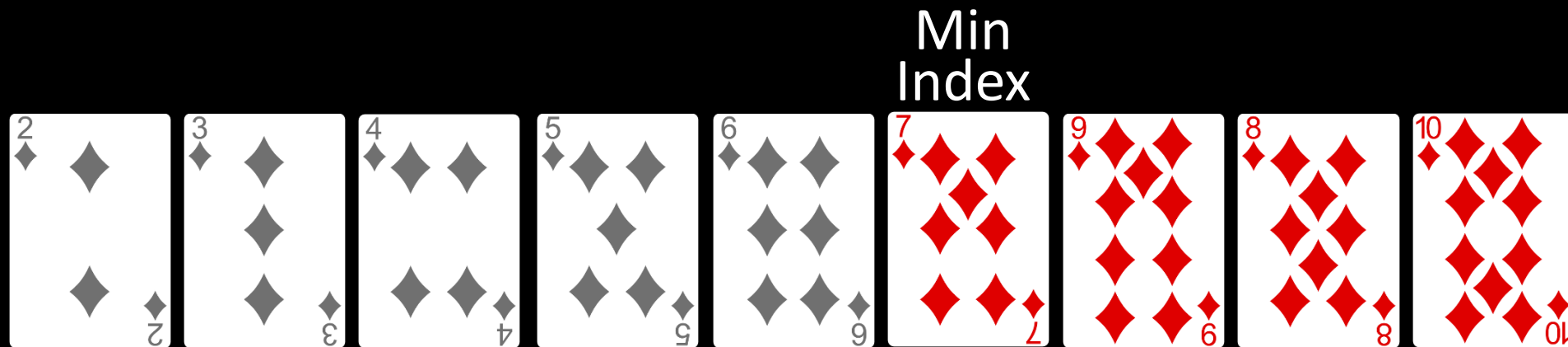
Пример за сортиране чрез пряка селекция



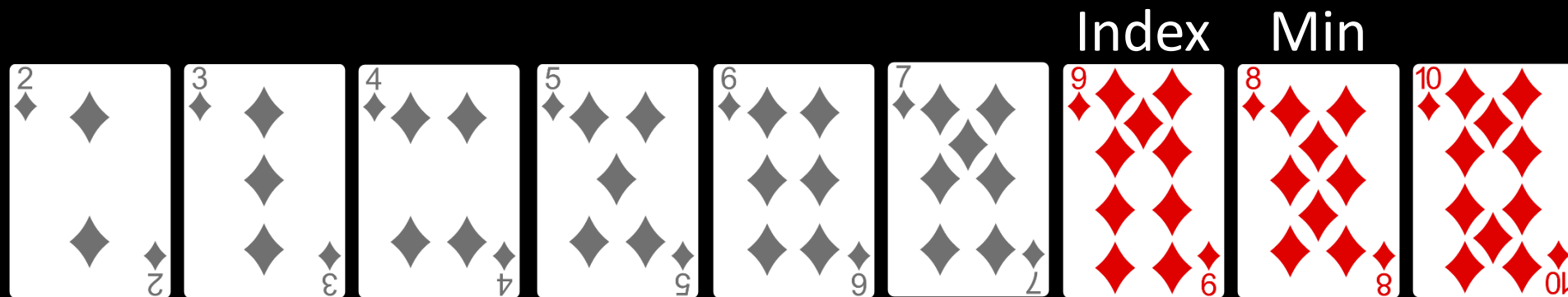
Пример за сортиране чрез пряка селекция



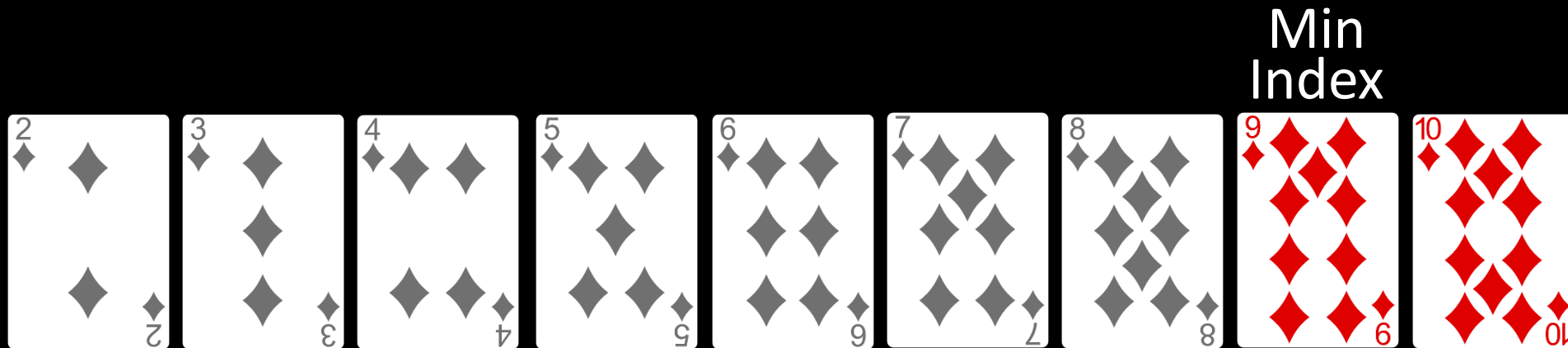
Пример за сортиране чрез пряка селекция



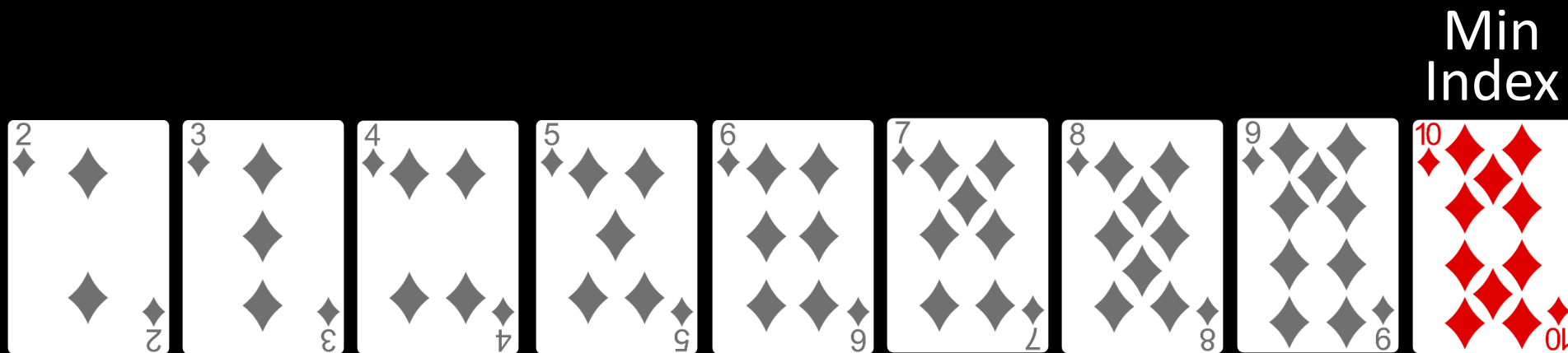
Пример за сортиране чрез пряка селекция



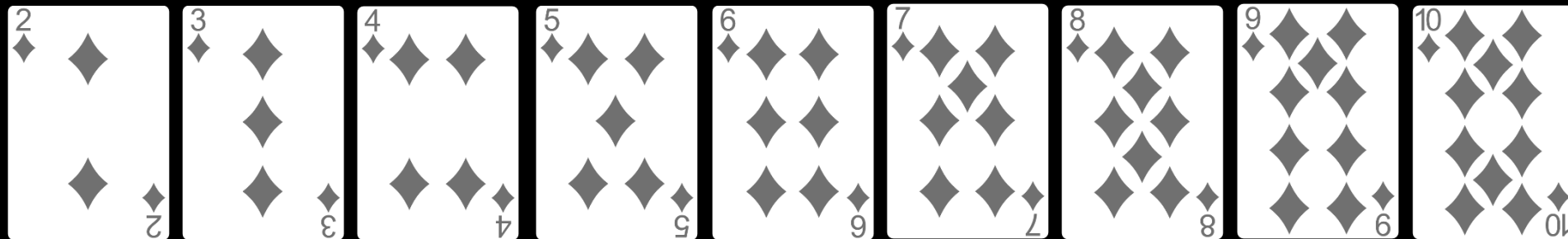
Пример за сортиране чрез пряка селекция



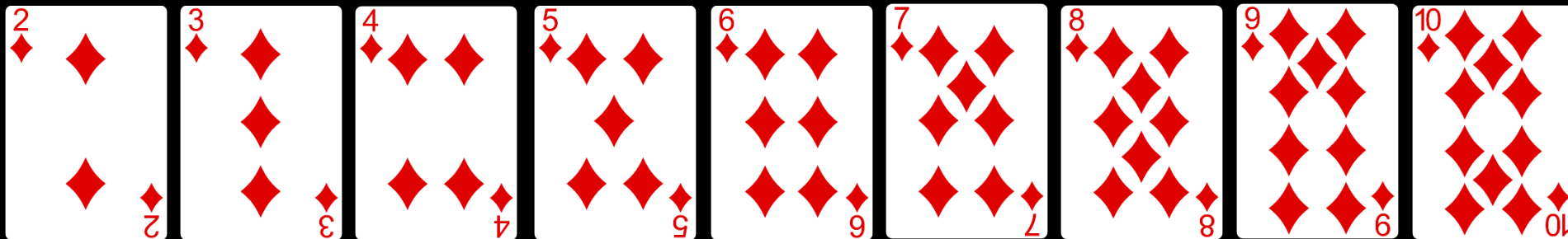
Пример за сортиране чрез пряка селекция



Пример за сортиране чрез пряка селекция



Пример за сортиране чрез пряка селекция



Сортиране чрез пряка селекция: защо е нестабилен?

- Защо „сортирането чрез пряка селекция“ е **нестабилно**?
 1. Разменя първият елемент с минималния елемент отдясно
 2. Разменя вторият елемент с минималния елемент отдясно
 3. ...
- По време на размените еднакви елементи се прескачат



Код за сортиране чрез пряка селекция

```
for (int index = 0; index < collection.Length; index++)
{
    int min = index;
    for (int curr = index + 1; curr < collection.Length; curr++)
    {
        if (Less(collection[curr], collection[min]))
        {
            min = curr;
        }
    }
    Swap(collection, index, min);
}
```

Сравнение на сортиращите алгоритми

Име	Най-добре	Средно	Най-зле	Памет	Стабилен	Метод
<u>SelectionSort</u>	n^2	n^2	n^2	1	Не	Селекция

Обобщение

- Сортиращите алгоритми подреждат елементи на списък
 - Елементите трябва да са сравними
- Алгоритмите се класифицират според
 - изчислителната сложност
 - използвания метод и памет
 - дали са стабилни
 - дали са рекурсивни
- Сортиране чрез пряка селекция (Selection sort)
 - прост, но неефективен и нестабилен алгоритъм



Въведение в сортирането



Въпроси?

Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма "**Обучение за ИТ кариера**" на МОН за подготовка по професия "Приложен програмист"



Министерство
на образованието
и науката



Национална
програма
„Обучение за
ИТ кариера“

- Курсът е базиран на учебно съдържание и методика, предоставени от **фондация "Софтуерен университет"** и се разпространява под свободен лиценз **CC-BY-NC-SA**



SoftUni
Foundation

