



Национална програма
"Обучение за ИТ умения и кариера"
<https://it-kariera.mon.bg>

Министерството на
образованието и науката
<https://www.mon.bg>

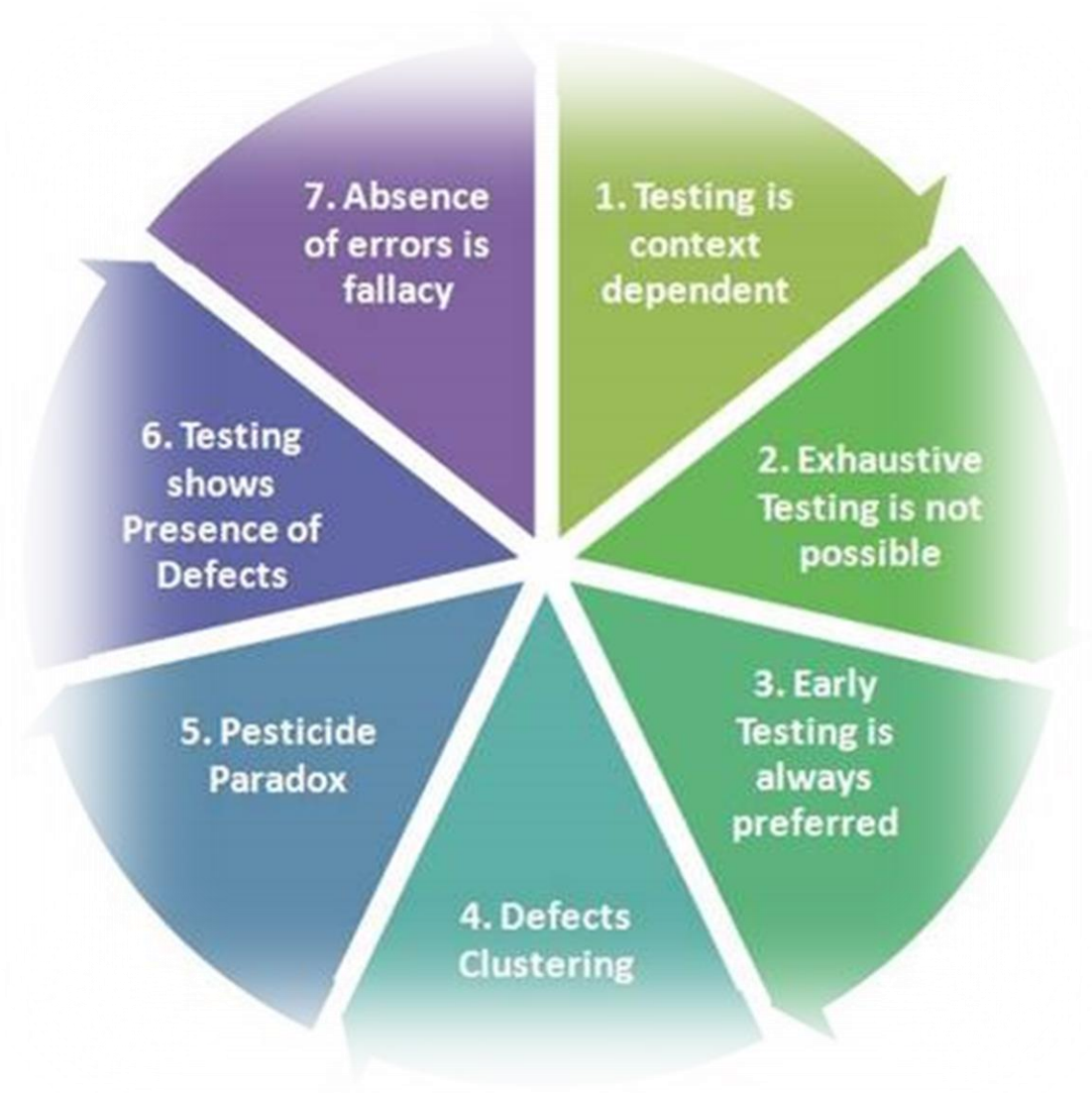


Компонентно тестване

Unit Testing

Съдържание

1. Седемте принципа на тестването
2. Какво е компонентното тестване?



Седемте принципа на тестването

Седемте принципа на тестването [1/7]

- Тестването е зависимо от контекста
 - Тестването се извършва по различни контексти
 - Пример: банков софтуер и текстов редактор се тестват по различен начин

Седемте принципа на тестването [2/7]

- Изчерпващото тестване е практически НЕВЪЗМОЖНО

- Всичките възможни комбинации от входни данни и начални условия обикновено се неизборимо много
- Тестването на всичко не е възможно
 - Освен на основните случаи
- Анализът на риска и приоритетите на софтуера показват къде трябва да насочим усилията при тестване

Седемте принципа на тестването [3/7]

- По-добре рано, отколкото късно!
 - Дейностите по тестването трябва да започнат възможно най-рано
 - И трябва да са насочени към конкретни цели
 - Колкото по-късно се открие даден бъг – толкова по-зле

Седемте принципа на тестването [4/7]

- Групиране на грешките

- Тестването трябва да е пропорционално
 - На очакваната и по-късно наблюдавана честота на дефектите на модулите
- Обикновено малък брой модули съдържат по-големия брой дефекти

Седемте принципа на тестването [5/7]

- Парадоксът на пестицидите

- Едни и същи тестове, които се повтарят отново и отново, губят своята ефективност
 - Незасечените досега дефекти остават неоткрити
- Трябва да се разработват нови и модифицирани тестове

Седемте принципа на тестването [6/7]

- Тестването показва наличието на дефекти, но не доказва отсъствието им!
 - Тестването показва съществуващи дефекти
 - Тестването не доказва отсъствието на други дефекти
 - Подходящото тестване намаля вероятността от дефекти

Седемте принципа на тестването [7/7]

- **Заблудата за липса на грешки**

- Намирането и оправянето на дефекти само по себе си не е достатъчно в тези случаи:
 - Продуктът е направен неудобно, неизползваемо и неоптимално
 - Продуктът не покрива изискванията, нуждите и очакванията на потребителите



Какво е компонентното тестване?

Софтуер, който тества софтуер

Ръчно тестване [1/2]

- Не е структурирано
- Не е повторимо
- Не покрива целия код
- Не е толкова лесно колкото би трябвало да бъде

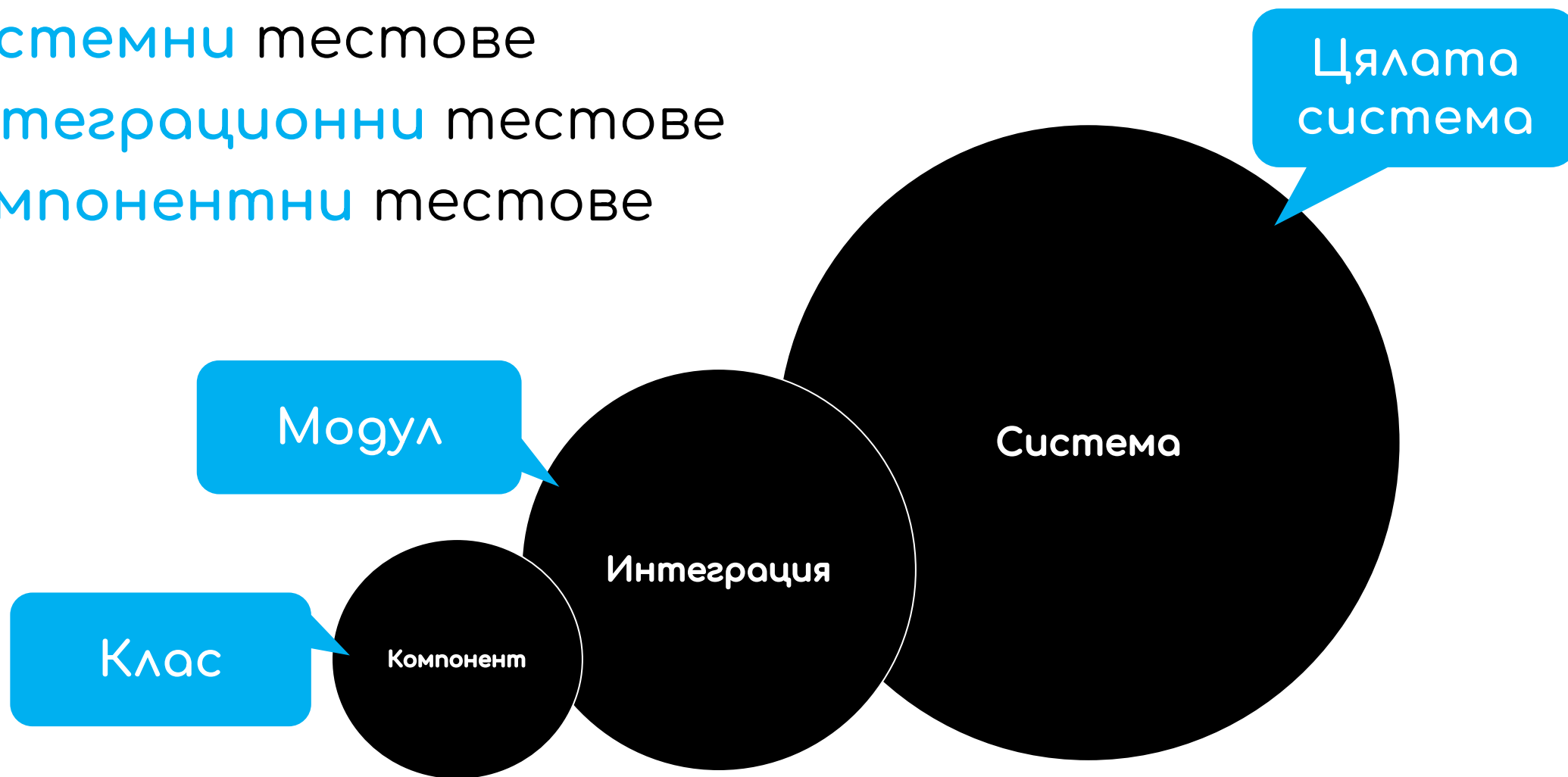
```
void testSum() {  
    if (this.sum(1, 2) != 3) {  
        throw new Exception("1 + 2 != 3");  
    }  
}
```

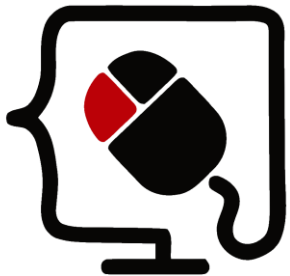
Ръчно тестване [2/2]

- Имаме нужда от **структуриран подход**, който:
 - Позволява **рефакториране**
 - Намаля цената за промяна
 - Намаля броя на **дефектите** в кода
- Бонус:
 - Подобрява проектирането

Автоматично тестване

- Системни тестове
- Интеграционни тестове
- Компонентни тестове





Национална програма
"Обучение за ИТ умения и кариера"
<https://it-kariera.mon.bg>

Министерството на
образованието и науката
<https://www.mon.bg>



Документът е разработен за нуждите на Национална програма "Обучение за ИТ умения и кариера" на Министерството на образованието и науката (МОН) и се разпространява под свободен лиценз CC-BY-NC-SA (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).