

## Поправителен изпит по ПИКЗ

Име: .....Ф№: ..... Група: .....

В научна конференция участват организатори, автори и гости. За всеки човек се пази име, адрес и телефон за контакт. Авторите могат да участват със самостоятелни статии или статии в колектив, като за всеки от тях се пази число – с колко статии участва в конференцията. За всяка статия се пази заглавие, анотация (кратко описание), до четири ключови думи и основен текст. За статиите в колектив се пази допълнително информация за това кой автор какво процентно участие има в нея. Всяка статия преминава през три етапа – нова (няма назначен рецензент), в процес на рецензия (има назначен рецензент, но още не е готова) и готова (което може да е „одобрена“ или „отхвърлена“). Рецензент може да е само човек от организаторите на конференцията. Организаторите също могат да бъдат автори, но един организатор-автор не може да е рецензент сам на себе си.

Моделирайте класове спрямо описаните обекти. При създаването на класовете демонстрирайте знания за наследяване, валидация и капсулиране на данни.

Само за организаторите добавете поле с парола (private), която ще се използва за верификация при промяна на статус на статия (виж описанието на метод `changeStatus` по-долу). Създайте метод `Boolean verifyPass(String)`, който по подадена парола връща отговор дали е вярна или не.

За статиите добавете следните методи:

- `String getAuthor() / getAuthors(int)` – връща името на автора (без параметър връща първия автор, с параметър връща 1ви, 2ри и т.н. автор за статиите в колектив);
- `void getStatus()` – извежда на екрана информация за текущия статус на статията (етап);
- `void changeStatus(...)` – по подадени параметри (вие избирате какви) променя статуса на дадена статия (добавя рецензент или одобрява/отхвърля статия, като е задължително един от параметрите да бъде парола, която ще се сравнява с паролата на рецензента за съвпадение).

Не дефинирайте собствени изключения, а там където е необходимо да се хвърли изключение използвайте основния клас `Exception`. Използвайте само конструктори с подадени параметри (не правете конструктори по подразбиране!).

Направете клас с `main` метод, в който демонстрирайте пример за полиморфизъм с някои от създадените класове.