Показатели, по которым оценивается здоровье кода

Пустая программа даёт 100%

Дающие отрицательный процент:

1. Количество var вне блоков. За первую - -10%, за остальные - -2% пока не достигнет 25%
2. Описание параметра цикла for вне заголовка - за первую - -15%, за каждую следующую - -3% пока не достигнет 25%
3. Использование процедуры Read для ввода - за первую -15%, за последующие - -2% пока не достигнет -20%
4. Использование ключевого слова program - -10%
5. Использование статических массивов - -10% за первый, за каждый последующий - -2% пока не достигнет -15%
6. Использование Write(a,’ ‘,b) с пробелом в качестве разделителя - за каждую - -1% пока не достигнет -5%
7. Использование устаревшего типа string[10] - за каждую - -1% . Если минусуется > 100%, то здоровье делать 0%

Дающие положительный процент:

1. Используются внутриблочные описания переменных: За первую - 10%, за остальные - 2% (максимум 4 раза)
2. Переменная цикла for описана в заголовке цикла: 3% за каждую (максимум 4 раза)
3. Используется инициализация переменной при описании: 4% за каждую (максимум 4 раза)
4. Для ввода используется функция вида ReadInteger, ReadReal: 3% за каждую (максимум 4 раза)
5. Используется расширенное присваивание += \*=: 3% за каждую (максимум 4 раза)
6. Для вывода использована Print: 4% за каждую (максимум 4 раза)
7. Используются кортежи: 5% за каждый (максимум 4 раза)
8. Используются динамические массивы: 5% за каждый (максимум 4 раза)
9. Используется распаковка значения в переменные: 5% за каждое (максимум 4 раза)
10. Используется цикл loop: 4% за каждый (максимум 4 раза)
11. Используется цикл foreach: 4% за каждый (максимум 4 раза)
12. Используются лямбда-выражения: 8% за каждый (максимум 4 раза)

Максимальное здоровье программы ограничено 200%