



Университет ИТМО
Факультет ПИ и КТ

Лабораторная работа №1

По Дисциплине «Тестирование программного обеспечения».



Выполнил: Дробыш Дмитрий
Александрович
Группа: Р33082
Преподаватель: Харитонов
Анастасия Евгеньевна

Лабораторная работа #1

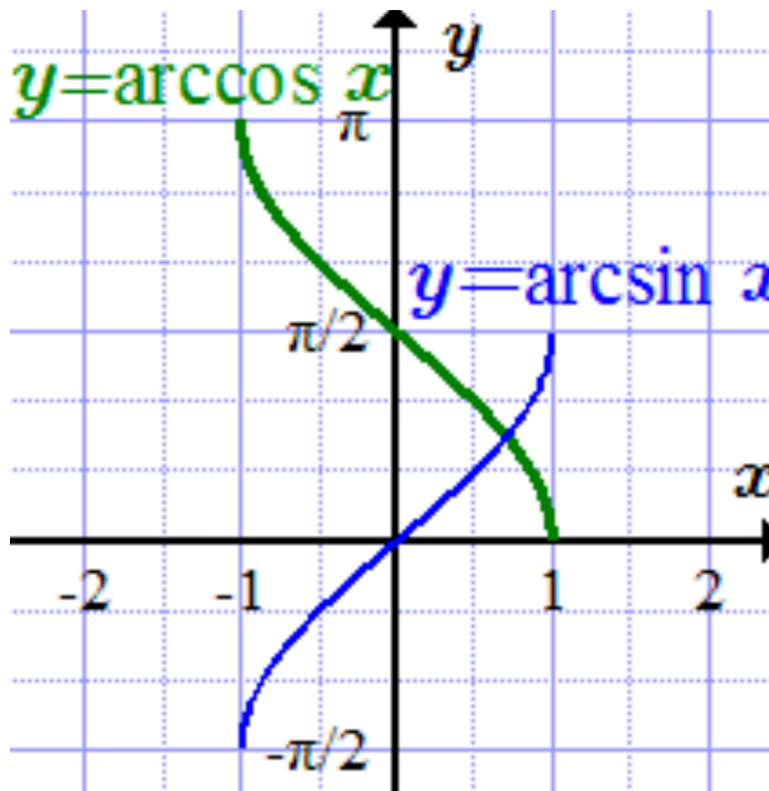
1. Для указанной функции провести модульное тестирование разложения функции в степенной ряд. Выбрать достаточное тестовое покрытие.
2. Провести модульное тестирование указанного алгоритма. Для этого выбрать характерные точки внутри алгоритма, и для предложенных самостоятельно наборов исходных данных записать последовательность попадания в характерные точки. Сравнить последовательность попадания с эталонной.
3. Сформировать доменную модель для заданного текста. Разработать тестовое покрытие для данной доменной модели

Введите вариант:

1. Функция $\arccos(x)$
2. Программный модуль для работы с бинарным деревом (<http://www.cs.usfca.edu/~galles/visualization/BST.html>)
3. Описание предметной области:

Они сидели на мостовой и смотрели с некоторым беспокойством, как огромные дети тяжело прыгают по песку, а дикие лошади с грохотом везут по небу в Неизведанные Области свежие запасы армированных изгородей.

1. ACOS X.



Функция $y = \arccos x$ является обратной к функции $y = \cos x$, где $0 \leq x \leq \pi$. График функции $y = \arccos x$ является симметричным графику функции $y = \cos x$, где $0 \leq x \leq \pi$.
 $\Rightarrow x \in [-1; 1]$

Причина выбора тестового покрытия :
`doubles = {1/2d, -1/2d, 0.9, -0.9}` — Algorithm validator.

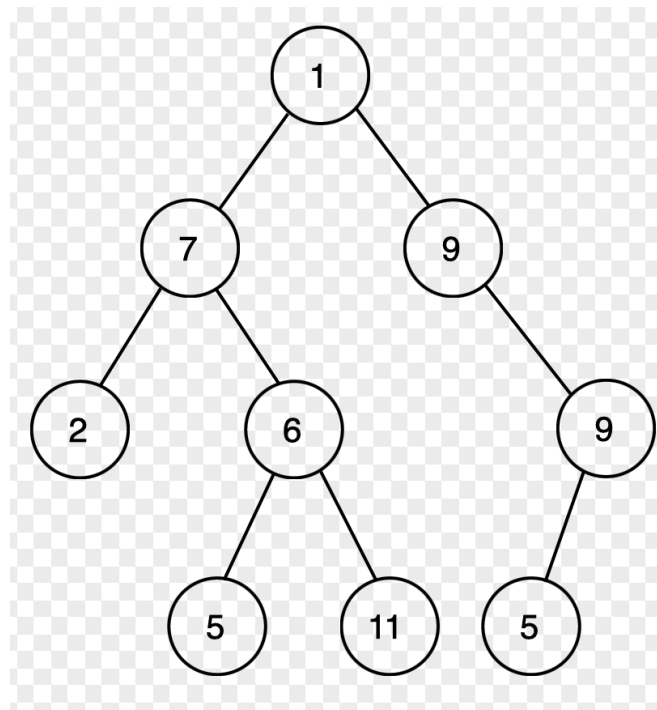
`doubles = {0, 1, -1}` - Edges and 0 point.

`ints = {0, -2, -1}` - incorrect n values. Waiting for error.

`doubles = {2, -2}` - incorrect x values. Waiting for error.

2. BTREE

Struct nodes -> Struct BTREE



createRootElement
addToRoot
addSomeElementsToRoot
deleteRoot
deleteTwoChildrenRoot
deleteTwoChildrenRootV2
deleteTwoChildrenRootV3
deleteOneChildRoot
deleteOneChildRootV2
deleteOneChildRootV3
deleteOneChildRootV4

-----> Testing working with root nodes. (Due to implementation)

addSameElements
sameElementsOnlyDeleteTest
sameElementsOnlyDeleteTestV2

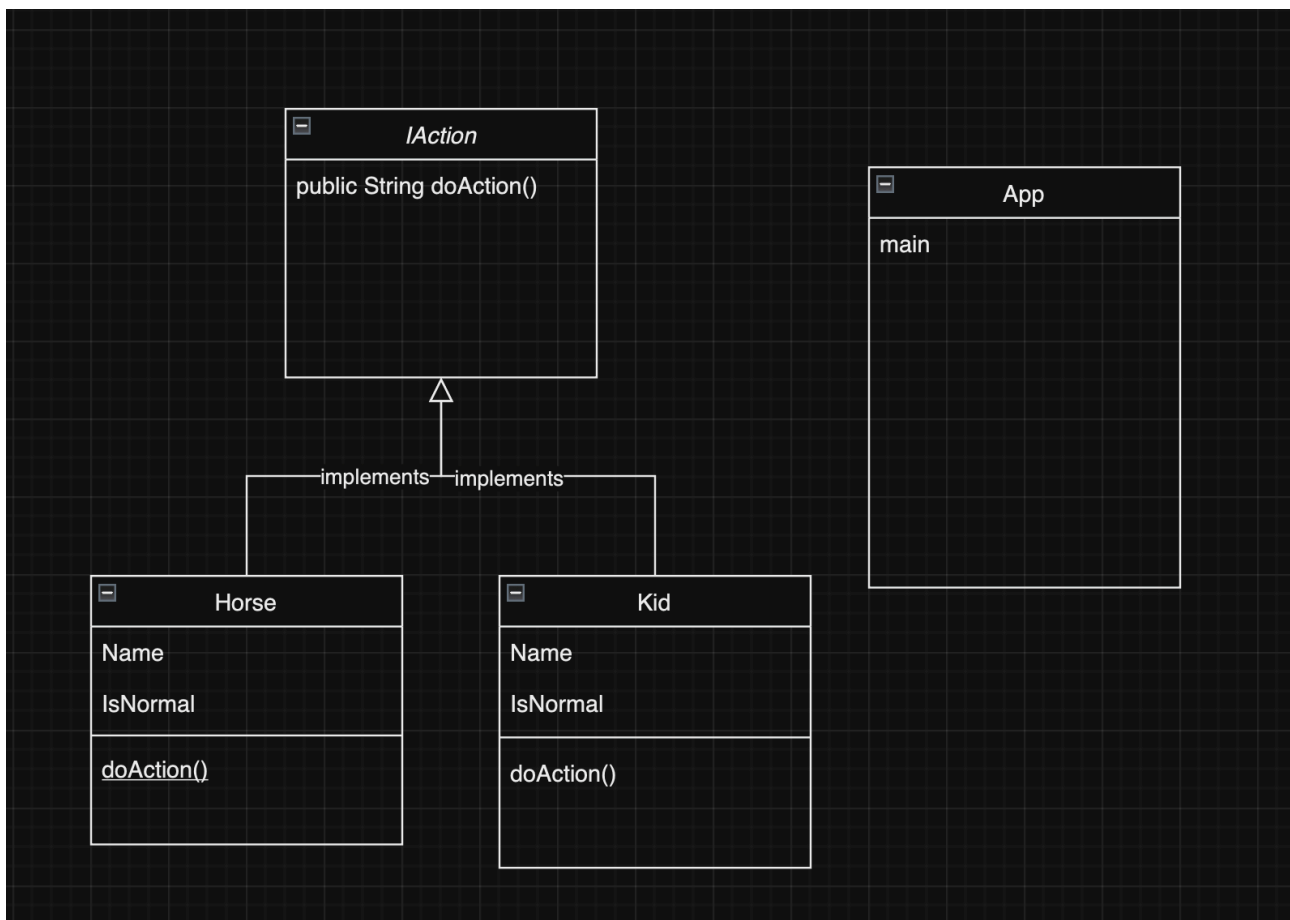
-----> Testing adding same elements (Due to implementation)

addSomeElementsToRegularNode
LeafDelete
OneChildDelete
TwoChildrenDelete

-----> Validator.

3. Story

- * Они сидели на мостовой и смотрели с некоторым беспокойством,
- * как огромные дети тяжело прыгают по песку, а дикие лошади
- * с грохотом везут по небу в Неизведанные Области
- * свежие запасы армированных изгородей.



CheckNameKid
CheckNormalKid
CheckDoActionKid

-----> Kid Validator.

CheckNameHorse
CheckNormalHorse
CheckDoActionHorse

-----> Horse Validator.

CheckNameWatcher
CheckLimitWatcher

-----> Watcher Validator and dependency check..

Вывод.

В ходе выполнения работы я познакомился с инструментом тестирования ПО JUNIT. Попробовал создавать как полностью ручные тесты, так и ParameterizedTest. Во время работы неоднократно тесты помогли выявить ошибки в работе кода, которые я пропустил на стадия кодирования.