

Лабораторная работа №4

Иванова Виктория Павловна 6203-010302D

Задание 1

В классах `ArrayTabulatedFunction` и `LinkedListTabulatedFunction` добавила конструкторы, получающие сразу все точки функции в виде массива объектов типа `FunctionPoint`. Если точек задано меньше двух, или если точки в массиве не упорядочены по значению абсциссы, конструкторы выбрасывают исключение `IllegalArgumentException`. Для обеспечения инкапсуляции создаются копии переданных точек.

Задание 2

В пакете `functions` создала интерфейс `Function`, содержащий методы `getLeftDomainBorder`, `getRightDomainBorder` и `getFunctionValue`. Интерфейс `TabulatedFunction` теперь расширяет `Function`. Соответствующие методы были удалены из `TabulatedFunction`, что убрало проблему дублирования.

Задание 3

Создала пакет `functions.basic`, в нём описала классы ряда функций. `Exp` реализует экспоненту с бесконечной областью определения, используя `Math.exp`. `Log` представляет логарифм с задаваемым основанием, проверяя его корректность в конструкторе. Для тригонометрических функций создана иерархия с базовым классом `TrigonometricFunction`, определяющим общую область определения, и производными классами `Sin`, `Cos` и `Tan`, использующими соответствующие методы из `Math`. Для получения значений воспользовалась методами `Math.sin()`, `Math.cos()` и `Math.tan()`.

Задание 4

Создала пакет `functions.meta`, в нём описала классы функций, позволяющие комбинировать функции. Создала класс `Sum`, объекты которого представляют собой функции, являющиеся суммой двух других функций, конструктор класса получает ссылки типа `Function` на объекты суммируемых функций, а область определения функции получается как пересечение областей определения исходных функций. Класс `Mult` реализован аналогично. Класс `Power` создает функцию, представляющую собой возведение значений исходной функции в заданную степень. Класс `Scale` реализует масштабирование функции вдоль осей координат. Значение функции рассчитывается как масштабированное по Y значение исходной функции от масштабированного по X аргумента. Коэффициенты масштабирования могут быть отрицательными, что позволяет осуществлять отражение функции относительно осей координат. Класс `Shift` выполняет сдвиг функции вдоль координатных осей. Вычисление значения функции производится как значение исходной функции от аргумента, сдвинутого на величину `shiftX`, плюс сдвиг по Y . Класс `Composition` реализует композицию двух функций. Вычисление значения происходит путем последовательного применения функций: значение первой функции в точке x передается как аргумент второй функции.

Задание 5

В пакете `functions` создала класс `Functions`, содержащий вспомогательные статические методы для работы с функциями. Класс содержит следующие методы:
`public static Function shift(Function f, double shiftX, double shiftY)` – возвращает объект функции, полученной из исходной сдвигом вдоль осей;

public static Function scale(Function f, double scaleX, double scaleY) – возвращает объект функции, полученной из исходной масштабированием вдоль осей; public static Function power(Function f, double power) – возвращает объект функции, являющейся заданной степенью исходной; public static Function sum(Function f1, Function f2) – возвращает объект функции, являющейся суммой двух исходных; public static Function mult(Function f1, Function f2) – возвращает объект функции, являющейся произведением двух исходных; public static Function composition(Function f1, Function f2) – возвращает объект функции, являющейся композицией двух исходных. Методы shift, scale, power, sum, mult, composition инкапсулируют создание объектов соответствующих классов, обеспечивая удобство и читаемость для пользователей системы.

Задание 6

Реализовала класс TabulatedFunctions с методом tabulate, который преобразует аналитическую функцию в табулированную на заданном отрезке. Метод проверяет, что границы табулирования не выходят за область определения функции, и выбрасывает IllegalArgumentException при нарушении этого условия. Возвращается интерфейсный тип TabulatedFunction, что обеспечивает гибкость выбора реализации.

Задание 7

В класс TabulatedFunctions добавила четыре метода для работы с потоками ввода-вывода. Метод outputTabulatedFunction выполняет запись табулированной функции в байтовый. Метод inputTabulatedFunction осуществляет чтение из байтового потока. Для работы с текстовыми потоками реализован метод writeTabulatedFunction, Метод readTabulatedFunction выполняет чтение из символьного потока с применением класса StreamTokenizer, который разбивает входной текст на токены. Метод последовательно считывает количество точек и затем пары координат, создавая на их основе табулированную функцию.

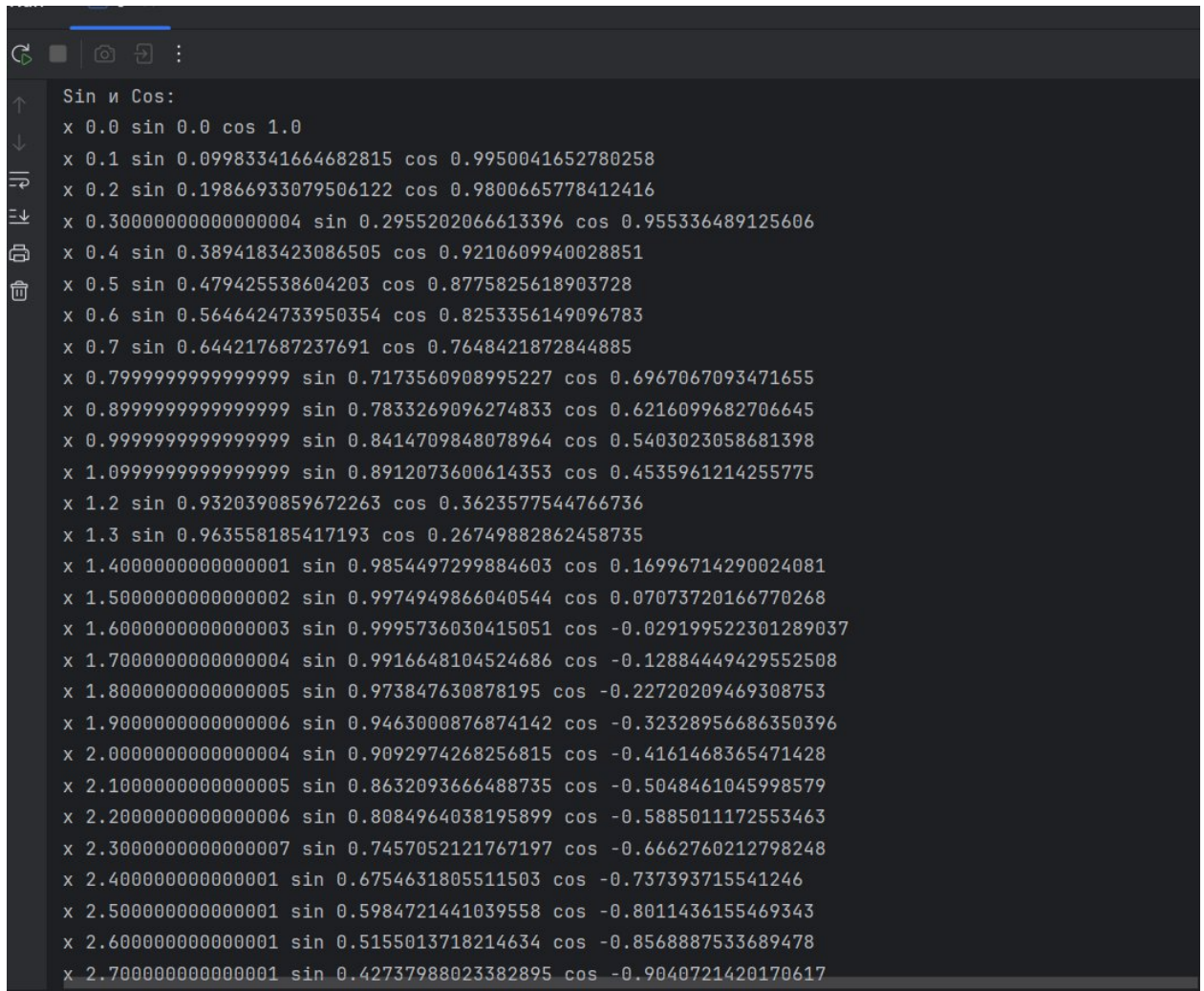
При проектировании методов пробрасывала исключения IOException наружу, чтобы предоставить вызывающему коду гибкость в обработке ошибок ввода-вывода согласно конкретному контексту использования. Потоки не закрываются внутри методов, так как управление жизненным циклом потоков должно оставаться за кодом, который их создал, что позволяет повторно использовать потоки для нескольких операций и избегать преждевременного закрытия.

Задание 8

Проверила работу написанных классов. Созданы и протестированы функции Sin и Cos, сравнены аналитические и табулированные значения. Исследована сумма квадратов $\sin^2 + \cos^2$, показавшая ожидаемое значение близкое к 1.0. Изучено влияние количества точек на точность табулирования - с увеличением точек от 5 до 40 точность возрастает. Протестирована работа с файлами для экспоненты и логарифма, подтверждена идентичность исходных и восстановленных данных.

Задание 9

Реализовала сериализацию двумя способами. Протестирована сериализация функции $\ln(\exp(x)) = x$, подтверждена полная идентичность исходных и десериализованных значений. Выявлены преимущества `Serializable` в простоте реализации и `Externalizable` в эффективности и контроле над процессом. Выбрала сериализацию `Externalizable`, тк у нее выше производительность, как записываются только нужные данные гибкость. Также она отличается от **`Serializable` ручным управлением** - сами контролируете что и как сериализуется.



```
Sin и Cos:
x 0.0 sin 0.0 cos 1.0
x 0.1 sin 0.09983341664682815 cos 0.9950041652780258
x 0.2 sin 0.19866933079506122 cos 0.9800665778412416
x 0.30000000000000004 sin 0.2955202066613396 cos 0.955336489125606
x 0.4 sin 0.3894183423086505 cos 0.9210609940028851
x 0.5 sin 0.479425538604203 cos 0.8775825618903728
x 0.6 sin 0.5646424733950354 cos 0.8253356149096783
x 0.7 sin 0.644217687237691 cos 0.7648421872844885
x 0.7999999999999999 sin 0.7173560908995227 cos 0.6967067093471655
x 0.8999999999999999 sin 0.7833269096274833 cos 0.6216099682706645
x 0.9999999999999999 sin 0.8414709848078964 cos 0.5403023058681398
x 1.0999999999999999 sin 0.8912073600614353 cos 0.4535961214255775
x 1.2 sin 0.9320390859672263 cos 0.3623577544766736
x 1.3 sin 0.963558185417193 cos 0.26749882862458735
x 1.4000000000000001 sin 0.9854497299884603 cos 0.16996714290024081
x 1.5000000000000002 sin 0.9974949866040544 cos 0.07073720166770268
x 1.6000000000000003 sin 0.9995736030415051 cos -0.029199522301289037
x 1.7000000000000004 sin 0.9916648104524686 cos -0.12884449429552508
x 1.8000000000000005 sin 0.973847630878195 cos -0.22720209469308753
x 1.9000000000000006 sin 0.9463000876874142 cos -0.32328956686350396
x 2.0000000000000004 sin 0.9092974268256815 cos -0.4161468365471428
x 2.1000000000000005 sin 0.8632093666488735 cos -0.5048461045998579
x 2.2000000000000006 sin 0.8084964038195899 cos -0.5885011172553463
x 2.3000000000000007 sin 0.7457052121767197 cos -0.6662760212798248
x 2.4000000000000001 sin 0.6754631805511503 cos -0.737393715541246
x 2.5000000000000001 sin 0.5984721441039558 cos -0.8011436155469343
x 2.6000000000000001 sin 0.5155013718214634 cos -0.8568887533689478
x 2.7000000000000001 sin 0.42737988023382895 cos -0.9040721420170617
```



```
Project ▾ Main.java x ArrayTabulatedFunction.java FunctionPoint.java FunctionPointIndexOutOfBoundsException.java
Run 0 x
x 2.7000000000000001 sin 0.42737988023382895 cos -0.9040721420170617
x 2.8000000000000001 sin 0.33498815015590383 cos -0.9422223406686585
x 2.9000000000000002 sin 0.23924932921398112 cos -0.9709581651495908
x 3.0000000000000003 sin 0.141120080598659 cos -0.9899924966004456
x 3.1000000000000004 sin 0.04158066243328916 cos -0.9991351502732795
Табелированные аналоги
x 0.0 sin 0.0 tab 0.0 cos 1.0 tab 1.0
x 0.1 sin 0.09983341664682815 tab 0.09798155360510165 cos 0.9950041652780258 tab 0.9827232084876878
x 0.2 sin 0.19866933079506122 tab 0.1959631072102033 cos 0.9800665778412416 tab 0.9654464169753757
x 0.30000000000000004 sin 0.2955202066613396 tab 0.29394466081530496 cos 0.955336489125606 tab 0.9481696254630635
x 0.4 sin 0.3894183423086505 tab 0.38590680571121505 cos 0.9210609940028851 tab 0.914354644443779
x 0.5 sin 0.479425538604203 tab 0.47207033789781927 cos 0.8775825618903728 tab 0.864608105941514
x 0.6 sin 0.5646424733950354 tab 0.5582338700844235 cos 0.8253356149096783 tab 0.8148615674392491
x 0.7 sin 0.644217687237691 tab 0.6439824415274444 cos 0.7648421872844885 tab 0.7646204979800333
x 0.7999999999999999 sin 0.7173560908995227 tab 0.707935358675545 cos 0.6967067093471655 tab 0.6884043792119047
x 0.8999999999999999 sin 0.7833269096274833 tab 0.7718882758236456 cos 0.6216099682706645 tab 0.6121882604437761
x 0.9999999999999999 sin 0.8414709848078964 tab 0.8358411929717461 cos 0.5403023058681398 tab 0.5359721416756473
x 1.0999999999999999 sin 0.8912073600614353 tab 0.8839933571281424 cos 0.4535961214255775 tab 0.45063345391435955
x 1.2 sin 0.9320390859672263 tab 0.9180219935851436 cos 0.3623577544766736 tab 0.35714054363391856
x 1.3 sin 0.963558185417193 tab 0.9520506300421447 cos 0.26749882862458735 tab 0.26364763335347763
x 1.4000000000000001 sin 0.9854497299884603 tab 0.984807753012208 cos 0.16996714290024081 tab 0.16993052093895483
x 1.5000000000000002 sin 0.9974949866040544 tab 0.984807753012208 cos 0.07073720166770268 tab 0.07043744393442489
x 1.6000000000000003 sin 0.9995736030415051 tab 0.984807753012208 cos -0.029199522301289037 tab -0.029055633070105086
x 1.7000000000000004 sin 0.9916648104524686 tab 0.984807753012208 cos -0.12884449429552508 tab -0.12854871007463503
x 1.8000000000000005 sin 0.973847630878195 tab 0.966204042925035 cos -0.22720209469308753 tab -0.22476145104951817
x 1.9000000000000006 sin 0.9463008876874142 tab 0.932175406468034 cos -0.32328956686350396 tab -0.3182543613299591
x 2.0000000000000004 sin 0.9092974268256815 tab 0.8981467700110329 cos -0.4161468365471428 tab -0.41174727161039987
x 2.1000000000000005 sin 0.8632093666488735 tab 0.8624409082617227 cos -0.5048461045998579 tab -0.5042718354168345
x 2.2000000000000006 sin 0.8084964038195899 tab 0.7984879911136221 cos -0.5885011172553463 tab -0.5804879541849632
```

```
Project ▾ Main.java x ArrayTabulatedFunction.java FunctionPoint.java FunctionPointIndexOutOfBoundsException.java
Run 0 x
x 2.2000000000000006 sin 0.8084964038195899 tab 0.7984879911136221 cos -0.5885011172553463 tab -0.5804879541849632
x 2.3000000000000007 sin 0.7457052121767197 tab 0.7345350739655215 cos -0.6662760212798248 tab -0.656704072953092
x 2.4000000000000001 sin 0.6754631805511503 tab 0.670582156817421 cos -0.737393715541246 tab -0.7329201917212208
x 2.5000000000000001 sin 0.5984721441039558 tab 0.594071569547527 cos -0.8011436155469343 tab -0.7941706620070894
x 2.6000000000000001 sin 0.5155013718214634 tab 0.5079080373609227 cos -0.8568887533689478 tab -0.8439172005093544
x 2.7000000000000001 sin 0.42737988023382895 tab 0.42174450517431844 cos -0.9040721420170617 tab -0.8936637390116193
x 2.8000000000000001 sin 0.33498815015590383 tab 0.3346977889881713 cos -0.9422223406686585 tab -0.9409837494179168
x 2.9000000000000002 sin 0.23924932921398112 tab 0.23671623538306957 cos -0.9709581651495908 tab -0.958260540930229
x 3.0000000000000003 sin 0.141120080598659 tab 0.1387346817779678 cos -0.9899924966004456 tab -0.9755373324425413
x 3.1000000000000004 sin 0.04158066243328916 tab 0.04075312817286608 cos -0.9991351502732795 tab -0.9928141239548535
Сумма квадратов
x 0.0 результат 1.0
x 0.1 результат 0.9753452893472049
x 0.2 результат 0.9704883234380686
x 0.30000000000000004 результат 0.9854291022725908
x 0.4 результат 0.9849684785101429
x 0.5 результат 0.9703975807827336
x 0.6 результат 0.975624427798983
x 0.7 результат 0.9993578909268825
x 0.7999999999999999 результат 0.9750730613812004
x 0.8999999999999999 результат 0.9705859765791769
x 0.9999999999999999 результат 0.9858966365208117
x 1.0999999999999999 результат 0.9845147652334687
x 1.2 результат 0.9703137486131723
x 1.3 результат 0.9759104767365345
x 1.4000000000000001 результат 0.9987226923395387
x 1.5000000000000002 результат 0.9748077439009694
x 1.6000000000000003 результат 0.9706905402060587
x 1.7000000000000004 результат 0.9863710812548067
```

```

x 0.1 результат 0.7000000000000001
x 1.0999999999999999 результат 0.9845147652334687
x 1.2 результат 0.9703137486131723
x 1.3 результат 0.9759104767365345
x 1.4000000000000001 результат 0.9987226923395387
x 1.5000000000000002 результат 0.9748077439009694
x 1.6000000000000003 результат 0.9706905402060587
x 1.7000000000000004 результат 0.9863710812548067
x 1.8000000000000005 результат 0.9840679624425679
x 1.9000000000000006 результат 0.9702368269293845
x 2.0000000000000004 результат 0.9762034361598597
x 2.1000000000000005 результат 0.9980944042379682
x 2.2000000000000006 результат 0.9745493369065118
x 2.3000000000000007 результат 0.9708020143187142
x 2.4000000000000001 результат 0.9868524364745752
x 2.5000000000000001 результат 0.9836280701374409
x 2.6000000000000001 результат 0.9701668157313703
x 2.7000000000000001 результат 0.9765033060689583
x 2.8000000000000001 результат 0.9974730266221714
x 2.9000000000000002 результат 0.974297840397828
x 3.0000000000000003 результат 0.9709203989171432
x 3.1000000000000004 результат 0.9873407021801173
Влияние количества точек на результат
Точек: 5, результат: 1.0
Точек: 10, результат: 0.9698463103929541
Точек: 20, результат: 0.9931806517013612
Точек: 40, результат: 0.9983786540671051

```

```

Точек: 40, результат: 0.9983786540671051
Экспонента
Сравнение экспоненты:
x 0.0 исходная 1.0 считанная 1.0
x 1.0 исходная 2.718281828459045 считанная 2.718281828459045
x 2.0 исходная 7.38905609893065 считанная 7.38905609893065
x 3.0 исходная 20.085536923187668 считанная 20.085536923187668
x 4.0 исходная 54.598150033144236 считанная 54.59815003314424
x 5.0 исходная 148.4131591025766 считанная 148.4131591025766
x 6.0 исходная 403.4287934927351 считанная 403.4287934927351
x 7.0 исходная 1096.6331584284585 считанная 1096.6331584284585
x 8.0 исходная 2980.9579870417283 считанная 2980.9579870417283
x 9.0 исходная 8103.083927575384 считанная 8103.083927575384
x 10.0 исходная 22026.465794806718 считанная 22026.46579480672
Логарифм
Сравнение логарифма
x 1.0 исходная 0.0 считанная 0.0
x 2.0 исходная 0.6931471805599453 считанная 0.6931471805599453
x 3.0 исходная 1.0986122886681098 считанная 1.0986122886681098
x 4.0 исходная 1.3862943611198906 считанная 1.3862943611198906
x 5.0 исходная 1.6094379124341003 считанная 1.6094379124341003
x 6.0 исходная 1.791759469228055 считанная 1.791759469228055
x 7.0 исходная 1.9459101490553132 считанная 1.9459101490553132
x 8.0 исходная 2.0794415416798357 считанная 2.0794415416798357
x 9.0 исходная 2.1972245773362196 считанная 2.1972245773362196
x 10.0 исходная 2.302585092994046 считанная 2.302585092994046

```



```

x 8.0 исходная 2.1972245773362196 считанная2.1972245773362196
x 9.0 исходная 2.1972245773362196 считанная2.1972245773362196
x 10.0 исходная 2.302585092994046 считанная2.302585092994046
ArrayTabulatedFunction (Serializable)
x 0: F = NaN, F1 = NaN
x 1: F = 1.0, F1 = 1.0
x 2: F = 2.0, F1 = 2.0
x 3: F = 3.0000000000000004, F1 = 3.0000000000000004
x 4: F = 4.0, F1 = 4.0
x 5: F = 4.999999999999999, F1 = 4.999999999999999
x 6: F = 6.0, F1 = 6.0
x 7: F = 6.999999999999999, F1 = 6.999999999999999
x 8: F = 7.999999999999998, F1 = 7.999999999999998
x 9: F = 9.000000000000002, F1 = 9.000000000000002
x 10: F = 10.000000000000002, F1 = 10.000000000000002
LinkedListTabulatedFunction (Externalizable)
x = 1 F = 1.0, F1 = 1.0
x = 2 F = 2.0, F1 = 2.0
x = 3 F = 3.0000000000000004, F1 = 3.0000000000000004
x = 4 F = 4.0, F1 = 4.0
x = 5 F = 4.999999999999999, F1 = 4.999999999999999
x = 6 F = 6.0, F1 = 6.0
x = 7 F = 6.999999999999999, F1 = 6.999999999999999
x = 8 F = 7.999999999999998, F1 = 7.999999999999998
x = 9 F = 9.000000000000002, F1 = 9.000000000000002
x = 10 F = 10.000000000000002, F1 = 10.000000000000002
Process finished with exit code 0
```