

Лабораторная работа №6

Иванова Виктория 6203-010302D

Задание 1

Разработала метод интегрирования по методу трапеций в классе Functions. Данный метод принимает в качестве параметров ссылку на объект функции, левую и правую границы интегрирования, а также шаг дискретизации. Особенностью реализации является проверка на выход интервала интегрирования за границы области определения функции с генерацией исключения в случае нарушения. Алгоритм метода трапеций заключается в разбиении области интегрирования на участки заданной длины с последующим вычислением площади под кривой как суммы площадей трапеций. Тестирование метода проводилось на примере функции экспоненты на отрезке $[0,1]$. Определила также, какой шаг дискретизации нужен, чтобы рассчитанное значение отличалось от теоретического в 7 знаке после запятой, 0.0001.

Задание 2

Создала пакет threads, в котором размещены классы, связанные с потоками. Создала класс Task в пакете threads, инкапсулирующий параметры задания: ссылку на интегрируемую функцию, границы области интегрирования, шаг дискретизации и количество заданий. Реализовала в мейне метод nonThread(), выполняющий последовательную обработку заданий в цикле. На каждой итерации генерируется логарифмическая функция со случайным основанием от 1 до 10, устанавливаются случайные границы интегрирования (левая от 0 до 100, правая от 100 до 200) и случайный шаг дискретизации от 0 до 1. После формирования параметров выводится сообщение "Source" с указанием значений границ и шага, затем вычисляется интеграл и выводится сообщение "Result" с результатом вычислений.

Задание 3

В пакете threads создала два класса. Создала классы SimpleGenerator и SimpleIntegrator, реализующие интерфейс Runnable. Для решения проблем NullPointerException и несогласованности данных применены блоки синхронизации на объекте task и добавлена проверка на null в цикле ожидания данных. В классфх реализована логика ожидания освобождения задания через метод wait(), ожидания появления данных для обработки с использованием notify() для координации работы потоков.

Задание 4

В пакете threads создала два следующих класса. Создала классы Generator и Integrator, расширяющие класс Thread. Первоначальная реализация с одним семафором не обеспечивала строгого чередования операций, поэтому архитектура переработана с введением двух семафоров для раздельного управления доступом генератора и интегратора. Это позволило достичь четкой последовательности операций. В методе complicatedThreads() реализовала прерывание рабочих потоков после 50 миллисекунд выполнения.

```
"C:\Program Files\Java\jdk-25\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2025.2.3\lib\
интеграл экспоненты
Теоретическое значение 1.718281828459045
Результат с вычислениями через метод трапеции с шагом 0.0001 1.7182818298909435
Разница 1.4318983776462346E-9
Точность 1e-7
Source 38.137562168658775 192.8680263935988 0.6808544879228313
Result 38.137562168658775 192.8680263935988 0.6808544879228313 323.2588019116304
Source 36.25883140252123 105.07115623697291 0.8845511444828559
Result 36.25883140252123 105.07115623697291 0.8845511444828559 131.002472164998
Source 39.165347882853716 134.96706266580156 0.13435981215715542
Result 39.165347882853716 134.96706266580156 0.13435981215715542 5514.846652970768
Source 60.20489112227951 115.44442223283741 0.9884858544993204
Result 60.20489112227951 115.44442223283741 0.9884858544993204 301.36566877005976
Source 8.471432396401557 117.01222453796905 0.3108620213627038
Result 8.471432396401557 117.01222453796905 0.3108620213627038 252.53100276548895
Source 67.40697175361501 140.45372185873435 0.1869205637422915
Result 67.40697175361501 140.45372185873435 0.1869205637422915 424.02820892937825
Source 5.189826147315724 120.34782592388007 0.45570668802748104
Result 5.189826147315724 120.34782592388007 0.45570668802748104 233.7243712469638
Source 50.65262625340552 165.31540503704008 0.9618799852250377
Result 50.65262625340552 165.31540503704008 0.9618799852250377 303.7921071314113
Source 34.977261765910264 193.75828229689597 0.6868136028930045
Result 34.977261765910264 193.75828229689597 0.6868136028930045 351.06725693756067
```

```
Source 25.315503273512974 128.29914582653313 0.7567568484059386 218.34520849574798
Result 25.315503273512974 128.29914582653313 0.7567568484059386 218.34520849574798
Source 33.09482488289335 153.11297418945298 0.48154990938325326
Result 33.09482488289335 153.11297418945298 0.48154990938325326 262.0722808343003
Source 91.86860844669114 137.91945400270686 0.49507460764047473
Result 91.86860844669114 137.91945400270686 0.49507460764047473 204.6810467408032
Source 19.30978247143955 169.28256565452384 0.3080605686353254
Result 19.30978247143955 169.28256565452384 0.3080605686353254 465.09900974602675
Source 43.16180501826765 164.48740184582715 0.5611322012175637
Result 43.16180501826765 164.48740184582715 0.5611322012175637 246.2303188162015
Source 50.02262820077598 199.57784572727425 0.9162208530395032
Result 50.02262820077598 199.57784572727425 0.9162208530395032 612.5312898565859
Source 86.5236384260828 169.96879709625375 0.05995484410945573
Result 86.5236384260828 169.96879709625375 0.05995484410945573 201.9729361322824
Source 55.3382421131164 172.08815405839715 0.15011191875333485
Result 55.3382421131164 172.08815405839715 0.15011191875333485 434.09797057231435
Source 68.08371294965853 105.5389280064683 0.12313414886823026
Result 68.08371294965853 105.5389280064683 0.12313414886823026 117.66239821489292
Source 39.93379215685942 134.43196072941464 0.8988678174399543
Result 39.93379215685942 134.43196072941464 0.8988678174399543 255.48658451990147
Source 36.63845990646828 113.82495043355514 0.6681064352316135
Result 36.63845990646828 113.82495043355514 0.6681064352316135 406.67120093558196
Source 13.651186065003863 182.6136146857847 0.12917768585609635
Result 13.651186065003863 182.6136146857847 0.12917768585609635 494.1045581735365
закончено

Process finished with exit code 0
```

