

Отчет по лабораторной работе №6

Студент: Стоколяс Юрий Юрьевич

Группа: 6201-120303D

Выполненные задания

Задание 1

1. Добавлен численный метод интегрирования (трапеции) в `functions/Functions.integrate()`.
2. Проверяются корректность шага и попадание пределов в область определения функции.
3. В `Main.task1()` вычислен интеграл `exp` на $[0;1]$; шаг уменьшается до достижения точности в 7-м знаке.

Задание 2

1. Реализован последовательный сценарий `Main.nonThread()`.
2. В каждой из ≥ 100 задач случайно выбираются основание логарифма, границы интегрирования и шаг.
3. На каждой итерации выводятся сообщения `Source ...` и `Result ...`, интеграл считается новым методом.

Задание 3

1. В пакете `threads` созданы `Task`, `SimpleGenerator`, `SimpleIntegrator`.
2. Потоки используют синхронизированные блоки при доступе к общему заданию, что исключает `NullPointerException` и смешивание данных.
3. Метод `Main.simpleThreads()` запускает оба потока и дожидается завершения.

Задание 4

1. Добавлен одноместный семафор `ReadWriteSemaphore` с разделением операций записи и чтения.
2. Потоки `Generator` и `Integrator` используют семафор вместо синхронизации для передачи задания.

3. В `Main.complicatedThreads()` потоки запускаются, через 50 мс прерываются методом `interrupt()`, обработка прерывания реализована в потоках.

Результаты тестирования

Запуск `Main` выполняет все этапы последовательно: подбор шага для `exp`, последовательные 100 задач, затем простые потоки и версия с семафором. На каждом шаге выводятся строки `Source ...` и `Result ...`, подтверждающие корректное формирование и обработку задач.

Фрагмент консольного вывода:

```
C:\Users\RusCreat\.jdks\graalvm-jdk-25\bin\java.exe ... Main
Интеграл exp на [0,1]: 1.7182818625982395, шаг: 4.8828125E-4
Точное значение: 1.718281828459045

Source 39.06109527452079 156.31954169071867 0.8405635658387679
Result 39.06109527452079 156.31954169071867 0.8405635658387679 788.2:
Source 43.206937228599344 106.71622842940015 0.213148952090219
Result 43.206937228599344 106.71622842940015 0.213148952090219 159.5:
Source 51.57867270438062 163.5628108026674 0.6627137182965686
Result 51.57867270438062 163.5628108026674 0.6627137182965686 329.69:
Source 84.102299467385 199.43211379111054 0.1745923064207323
Result 84.102299467385 199.43211379111054 0.1745923064207323 343.943:
Source 73.5880962942986 118.23400813254983 0.37235297415721513
Result 73.5880962942986 118.23400813254983 0.37235297415721513 147.9:
```

...

...

```
Source 59.978293785064594 159.1574941730996 0.19624571185986606
Result 59.978293785064594 159.1574941730996 0.19624571185986606 256.:
Source 89.17062519729784 139.53346951062974 0.9857258312172914
Result 89.17062519729784 139.53346951062974 0.9857258312172914 196.5:
Source 87.81322669622092 149.25292164664447 0.10098058413813127
Result 87.81322669622092 149.25292164664447 0.10098058413813127 148.:
```

Source 21.19058952680997 119.09403839460597 0.9667564624267146

Result 21.19058952680997 119.09403839460597 0.9667564624267146 178.0!

Source 41.110967850072306 172.85719801391429 0.25269652833298006

Result 41.110967850072306 172.85719801391429 0.25269652833298006 465