Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Programming Techniques for Scientific Simulations Exercise 8

HS 12
Prof. M. Troyer

Problem 8.1 Simpson integration with virtual functions (Block B)

Implement a new version of your Simpson integration routine using virtual functions.

- Define an abstract base class with a pure virtual `operator()` (function object), and a fully implemented simpson integrator, both for double precision.
- Inherit a concrete class from this abstract base class and implement the `operator()`.

Since the main part of Report B will be the benchmarking of all your Simpson integration routines, you can already start introducing timers in your implementations.

Problem 8.2 Factory design pattern

Virtual functions introduce runtime polymorphism, that we can use to create a factory for sorting algorithms.

1. Look at the skeleton code provided on the lecture homepage and complete it.
2. Use the factory pattern in your Simpson program, to allow the user to choose the function to be integrated.

Problem 8.3 Penna Model with Fishing (Block C)

(*S. Moss de Oliveira et al., Physica A 215, 298 (1995)*[1])

Implement a Penna fish simulation to observe how a slight increase of fishing may destroy an initially stable population. The following is how you should modify your original Penna simulation:

1. At a time-step M1, when the fish population is stable, introduce the concept of fishing. Here, any fish can die due to fishing (in addition to illness) with probability p_1. At a later time-step M_2 ($M_2 > M_1$) we increase the fishing to p_2 ($p_2 > p_1$).
2. Observe that an increase in fishing (ie. with $p_2 > p_1$ at time M_2) will destabilize the fish population.
3. Instead of (2) above, observe what will happen if fishing is only allowed for the adult species after time M1.

[1] `http://www.sciencedirect.com/science/article/pii/037843719500039A`