Проект

по Системи за Паралелна Обработка

Паралелен Алгоритъм за

Изчисляване на ПИ

Изготвил:

Николай Ненков

Ръководител:

ас. Христо Христов

Проверил: ........................

(ас. Христо Христов)

1. *Постановка на задачата*

Числото (стойността на) Pi може да бъде изчислена по различни начини. Разглеждаме

следния метод за приближено пресмятане на Pi:

1) Имаме окръжност, вписана в квадрат;

2) Генерираме по произволен начин точки в квадрата;

3) Определяме броя на точките, които се намират в окръжността;

4) Нека **k** е число равно на броя на точките в окръжността, разделен на броя на точките в

квадрата;

5) Тогава **Pi ~ 4 \* k**;

Бележка: Колкото повече точки генерираме толкова по-добра ще бъде точността с която

пресмятаме Pi.

Вашата задача е да напишете програма за изчисление на числото Pi. по-описания метод,

която използва паралелни процеси (нишки). Изискванията към програмата са следните:

(o) Размерността на квадрата се задава в точки (пиксели) от подходящо избран команден

параметър – например **“-s 10240”**. Точките в квадрата, генерираме произволно с помощта на

**Math.random()** или класа **java.util.Random;**

(о) Друг команден параметър задава максималния брой нишки (задачи) на които разделяме

работата по пресмятането на Pi – например **“–t 1”** или **“–tasks 3”;**

(о) Програмата извежда подходящи съобщения на различните етапи от работата си, както и

времето отделено за изчисление и резултата от изчислението (стойността на Pi);

(o) Да се осигури възможност за „quiet“ режим на работа на програмата, при който се

извежда само времето отделено за изчисление на Pi (и самото число) , отново чрез подходящо

избран друг команден параметър – например **“-q”;**

ЗАБЕЛЕЖКА:

(о) При желание за направата на подходящ графичен потребителски интерфейс (GUI) с

помощта на класовете от пакета **javax.swing** задачата може да се изпълни от **двама души**;

(о) Задачата може да се реши и с помощта на RMI (**java.rmi**). За целта трябва да се

помисли за разпределения достъп до общия ресурс в случая квадрата съдържащ случайно

генерираните точки;

1. *Описание на реализирания алгоритъм*

За паралелното решаване на задачата е избран подходът на многонишковото програмиране. Тъй като всяка точка се използва за единствена сметка – изчислението на разтоянието от центъра на кръга, те могат да бъдат генерирани при поискване. Използването на статичен Random генератор би довело до монолитен ресурс, следователно всяка нишка притежава свой собствен с чиято помощ да бъдат генерирани координатите на точките. Всеки от тези генератори е създаден със случайно генериран сийд, тъй като в противен случай биха се създали с еднакъв сийд давайки еднакви резултати. Общият брой точки се разделя м/у нишките и всяка изчислява броя на тези в кръга.

1. *Тестови замервания и анализ на метричните показатели*

Разработено е приложение на JAVA и е тествано на 8 ядрен мултипроцесор с цел да се оценят ускорението S (забързване, speed-up) и ефективността Е (efficiency) на описания алгоритъм, където ако T(p) е времето необходимо за завършване на работата на алгоритъм с p на брой нишки то:

S(p) = T(1)/T(p)

E(p) = S(p)/p

Замерванията са направени за 100000000 точки.