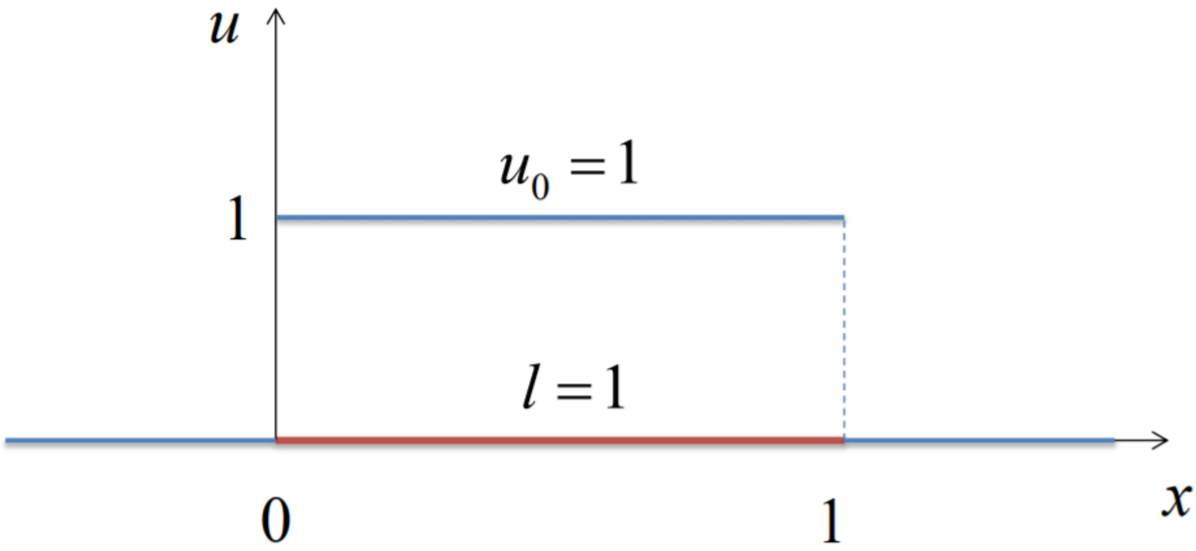
Домашнее задание №3 (MPI)

Задача о распараллеливание уравнения теплопроводности

Постановка задачи

Решить одномерное однородное уравнение теплопроводности с использование средств распараллеливания MPI.



Стержень длиной $l = 1$ в момент времени t_0 = 0 имеет температуру

u_0 = 1. Температура окружающей среды поддерживается равной 0.

Начальное условие: $u(x \in l, 0) = u_0$.

Граничное условие: $u(x \notin l, t) = 0$.

Начальное распределение температуры задаётся на 0 процессе и в конце алгоритма распределения температуры собираются на 0 процесс.

Необходимо решить одномерное однородное уравнение теплопроводности, которое представлено в виде конечно-разностной схемы:

$$u_i^{n+1} = u_i^n + \frac{k\tau}{h^2}\left(u_{i+1}^n - 2u_i^n + u_{i-1}^n\right)$$

$$\frac{k\tau}{h^2} < 1 \Rightarrow \tau < \frac{h^2}{k}$$

Задания

1) проверка правильной работы алгоритма: реализовать параллельную версию алгоритма и получить распределение температуры вдоль стержня на момент времени T = 0.1, используя следующие параметры: k = 1 (коэффициент температуропроводности), h = 0.02 (шаг по пространству); τ = 0.0002 (шаг по времени). Вычислить значения температуры в 11-ти точках длиной 0.1 включая начало (0, 0.1, 0.2, ..., 1) и сравнить полученные значения с точным решением:

$$u(x,t) = \frac{4u_0}{\pi}\sum_{m=0}^{\infty}\frac{exp\left(-k\pi^2(2m+1)^2\frac{t}{l^2}\right)}{2m+1}\sin\left(\frac{\pi(2m+1)x}{l}\right)$$

2) сравнить время работы программы для (1,2,4,8,16,24) процессов и разного кол-ва точек (N = 10000, 25000, 50000, конечное время можно уменьшить до 10^{-4} для уменьшения времени работы, условие для величины шага (τ) приведены выше), построить 3 графика соответственно кол-ву точек.

3) использовать коллективные операции для рассылки начального распределения температур и сбора итоговых результатов

Примечание по распараллеливанию: разумно будет поделить весь отрезок на NumberOfProc частей, чтобы каждый процессор считал свою часть (N/ NumberOfProc), при нахождении на границе между процессами воспользоваться передачей граничных значений “соседнему” процессору.

Дополнение

$$u_i^{n+1} = u_i^n + \frac{k\tau}{h^2}\left(u_{i+1}^n - 2u_i^n + u_{i-1}^n\right)$$

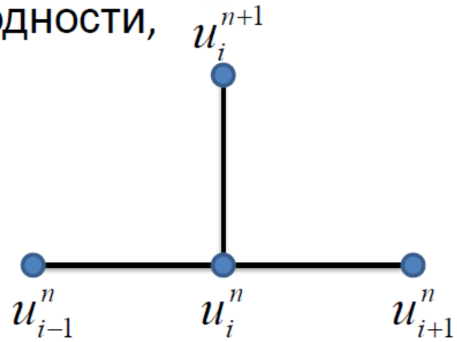
$$\frac{k\tau}{h^2} < 1 \Rightarrow \tau < \frac{h^2}{k}$$

где:

k - коэффициент температуропроводности,

τ - шаг по времени,

h - шаг по пространству



Задача состоит в нахождении распределения температур в момент времени n+1, таким образом нижний индекс – это индекс по пространству, а верхний индекс – это индекс по времени.

Как работает формула?

Пусть у нас есть начальный момент времени n = 0, в котором у нас задано распределение температуры ($u_{i\in l}{}^0 = 1, u_{i\notin l}{}^0 = 0$) и мы хотим найти распределение в следующий

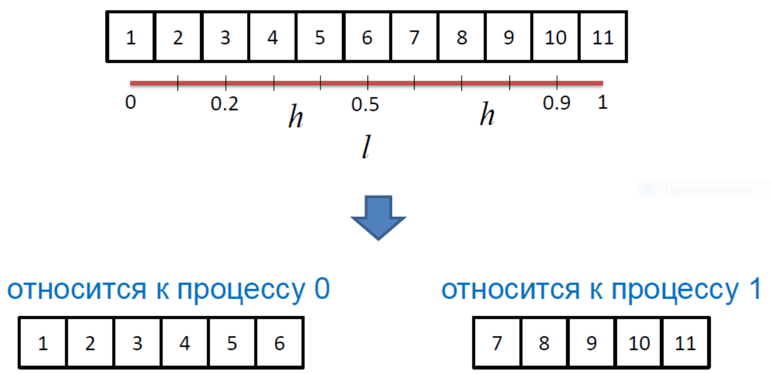
момент времени ($u_{i \in l}{}^1$), то есть в момент времени n+1 = τ.
Рассмотрим самую первую точку i = 0:

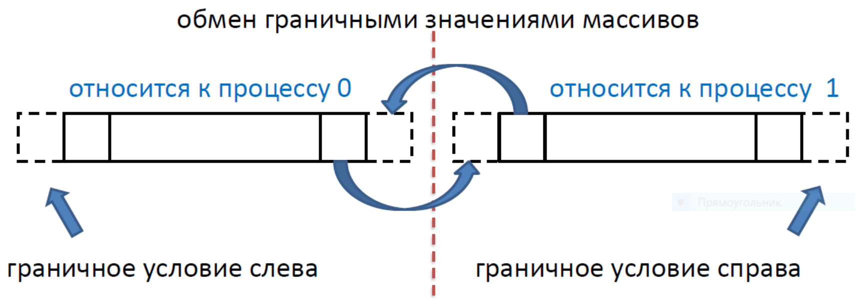
$$u_0{}^1 = u_0{}^0 + \frac{k\tau}{h^2}(u_1{}^0 - 2u_0{}^0 + u_{-1}{}^0)$$

$$u_0{}^1 = 1 + 0.5(1 - 2 + 0)$$

$$u_0{}^1 = 0.5$$

Таким образом мы получили значение в следующий момент времени для точки i = 0.
Как это работает параллельно? Допустим у нас есть 2 процесса и мы разделили весь стержень на 11 точек, соответственно 0-ой процесс будет обрабатывать 6 точек, а 1-ый процесс 5 точек. При расчёте граничных точек нужно воспользоваться передачей с предыдущего или следующего процесса, чтобы получить значение в предыдущей или следующей точке соответственно. В данном примере нужно было бы при расчёте 0 процессом в следующий момент времени в точке i = 6 передавать с 1-ого процесса значение в точке i = 7.





Оценивание

Задание 1: 4 балла

Задание 2: 4 балла

Задание 3: 2 балла

Сдача будет происходить очно на семинаре.
После сдачи задания вам нужно будет его прикрепить в [GoogleClassroom](). Файл программы нужно называть “heat_%Фамилия%_%группа%.cpp”,
например “heat_Ivanov_211.cpp”.
Файл с графиком “heat_%Фамилия%_%№ группы%.pdf”,
например “heat_Ivanov_211.pdf”

Дедлайн

Проверка задания будет 13.12.2022