Параллельное программирование (параллельные алгоритмы)

Лисицын Сергей
ФРКТ МФТИ 2019 г.

Программа 1 семестра

613 группа

Занятия: 06.02, 20.02, 6.03, 20.03, 03.04, 17.04, 1.05

Зачёт: 15.05

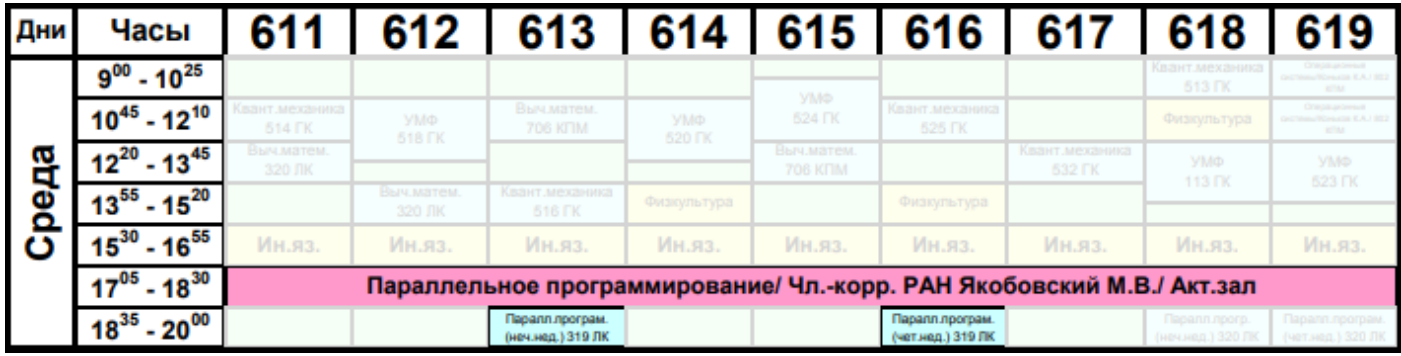
616 группа

Занятия: 13.02, 27.02, 13.03, 27.03, 10.04, 24.04, 8.05

Зачёт: 22.05

Месяц	Число месяца																														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Сентябрь		В							В							В							В						В	
Октябрь							В							В							В						В			
Ноябрь				*	В						В							В						В						
Декабрь		В							В							В	З	З	З	З	З	З	В	Э	Э	Э	Э	Э	Э	В	Э
Январь	*	*	*	*	*	*	*	*	Э	Э	Э	Э	В	Э	Э	Э	Э	Э	Э	В	Э	Э	Э	Э	Э	Э	В				
Февраль			В							В							В						*	В							
Март			В					*		В							В							В							В
Апрель							В							В							В						В			
Май	*				В				*			В			З				В	З	З	З	З	З	З	В	Э	Э	Э	Э	Э
Июнь	Э	В	Э	Э	Э	Э	Э	Э	В	Э	Э	*	Э	Э	Э	В	Э	Э	Э	Э	Э	Э	В	Э	Э	Э	Э	Э	Э	В	
Июль																															
Август																											П	П	П	П	П

Программа 1 семестра



Дни	Часы	611	612	613	614	615	616	617	618	619
Среда	9^{00} - 10^{25}								Квант.механика 513 ГК	
	10^{45} - 12^{10}	Квант.механика 514 ГК	УМФ 518 ГК	Выч.матем. 706 КПМ	УМФ 520 ГК	УМФ 524 ГК	Квант.механика 525 ГК		Физкультура	
	12^{20} - 13^{45}	Выч.матем. 320 ЛК				Выч.матем. 706 КПМ		Квант.механика 532 ГК	УМФ 113 ГК	УМФ 523 ГК
	13^{55} - 15^{20}		Выч.матем. 320 ЛК	Квант.механика 516 ГК	Физкультура		Физкультура			
	15^{30} - 16^{55}	Ин.яз.	Ин.яз.	Ин.яз.	Ин.яз.	Ин.яз.	Ин.яз.	Ин.яз.	Ин.яз.	Ин.яз.
	17^{05} - 18^{30}	Параллельное программирование/ Чл.-корр. РАН Якобовский М.В./ Акт.зал								
	18^{35} - 20^{00}			Парал.програм. (неч.нед.) 319 ЛК			Парал.програм. (чет.нед.) 319 ЛК		Парал.програм. (неч.нед.) 320 ЛК	Парал.програм. (чет.нед.) 320 ЛК

6 семинаров*:

1. Вводная лекция
2. Программирование на MPI
3. Статическая балансировка
4. Динамическая балансировка
5. Стандарт POSIX Threads
6. Программирование на общей памяти

*план может быть изменён

Программа 1 семестра

Программы:

- Обязательные задачи (4)
- Бонусные задачи (3)

Оценка за семестр:

+3: Лекционная контрольная

+2: Посещения/3

+2: Мгновенные обязательные задачи/2

+3: Бонусные задачи

Введение

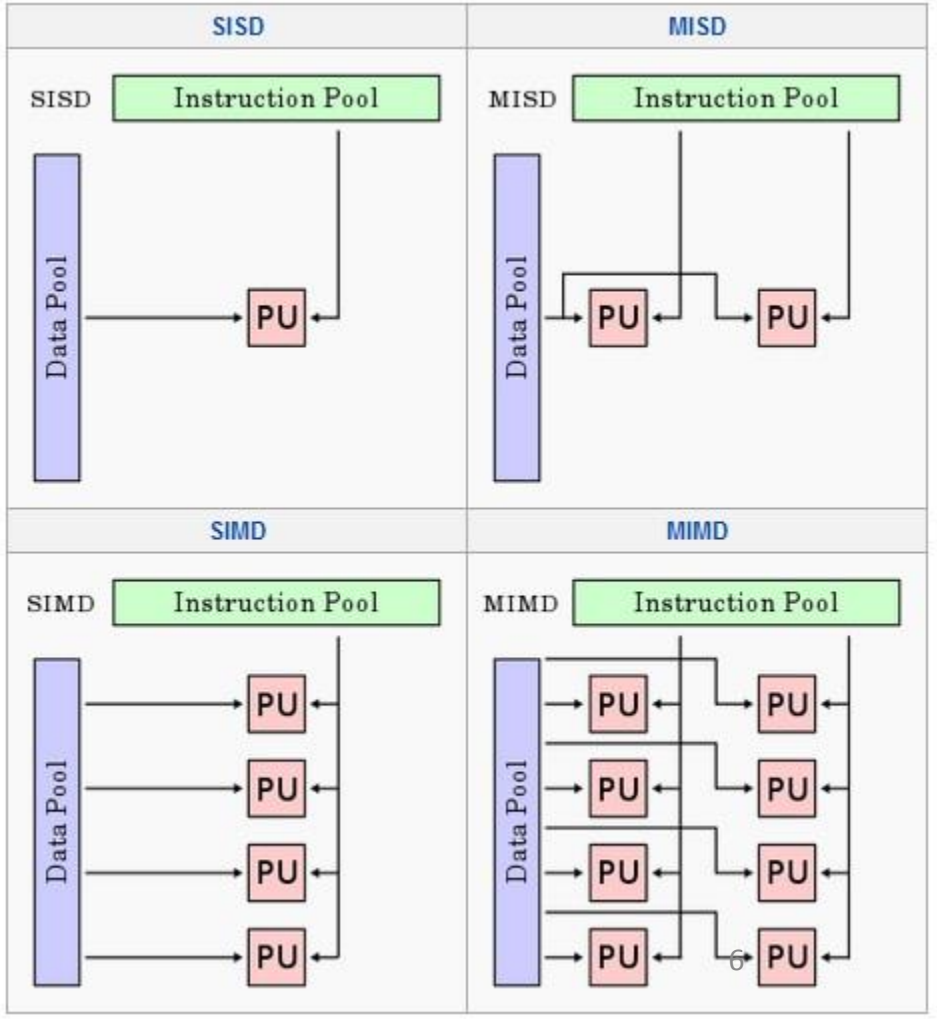
Perf = Freq*IPC/IC

- Конвейеризация вычислений (1970-е) – микроуровневый параллелизм
- Дублирование вычислителей (1980-е) – параллелизм уровня команд (векторизация, VLIW)
- Дублирование “конвейеров” (2000-е)- параллелизм уровня потоков/заданий

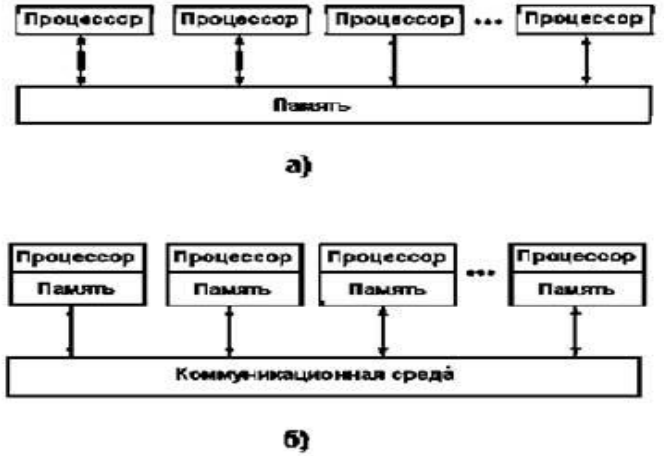
Введение

Таксономия (Классификация) Флинна (1966)

- SISD: компьютер фон-Неймановской архитектуры
- SIMD: векторные процессоры (MMX, SSE), матричные процессоры и процессоры с архитектурой VLIW.
- MISD: не используется
- MIMD:
  - Общая память - Symmetric Multiprocessor SMP
    Разделенная память - Massively Parallel Processing MPP -> Кластерные системы



Введение



Симметричное мультипроцессирование

1. Несколько однородных процессоров и массив общей памяти
2. Когерентность кэшей, урегулирование доступа к памяти
3. Ограниченная масштабируемость
4. Работает под единой ОС
5. Модель программирования: Потоки (pthread, OpenMP)

Массивно-параллельные системы

1. Вычислительные узлы и коммуникационная среда
2. Закрытая локальная память
3. Масштабируемость ~не ограниченная
4. Полная ОС на управляющей машине, на узлах урезанная версия
5. Модель программирования: Модель передачи сообщений (“fork”, MPI, PVM, BSPlib)

Введение

Метрики параллелизма

- Доля последовательных операций: $\alpha = \frac{IC_{|}}{(IC_{|} + IC_{||})}$
- Время на p потоках: $T_p = \alpha T_1 + \frac{(1-\alpha)T_1}{p}$
- Ускорение: $S = T_1/T_p$
- Эффективность: $E = S/p$

Закон Амдала

$$S = \frac{T_1}{T_p} = \frac{T_1}{\alpha T_1 + \frac{(1-\alpha)T_1}{p}} \leq \frac{1}{\alpha}$$

MPI

Message Passing Interface

Параллельная программа - множество одновременно выполняемых **процессов.**

Каждый процесс порождается на основе одного и того же программного кода (fork).

Количество процессов определяется в момент запуска программы.

Все процессы последовательно пронумерованы от 0 до **p-1** (ранг процесса), где **p** есть общее количество процессов.