**Программа курса «Параллельные вычисления»**

1. Основные определения. МВС. Суперкомпьютер, облако, кластер, грид-система, HPC(eng) и др. Классификации архитектур вычислительных систем. Классификации Флинна, Ванга-Бриггса, Фенга, Шора, Хендлера, Хокни, Скилликорна.

2. Архитектуры SMP, MPP. Архитектуры NUMA, PVP.

3. Кластерная архитектура. Особенности применения. Кластер типа Beowulf.

4. Особенности организации памяти в современных персональных компьютерах и МВС. Различные виды памяти. Иерархия памяти. Кэш-память.

5. Механизмы и алгоритмы обеспечения когерентности в МВС. Различные архитектуры МВС по типу доступа к памяти. (UMA. NUMA, NORMA и т.д.) Классификация архитектур. Общая схема.

6. Графические ускорители. Особенности организации памяти и вычислений.

7. Архитектура MIC. Особенности организации памяти и вычислений.

8. Топологии сетей МВС. Сферы применения. Свойства. Характеристики.

9. Способы оценки производительности МВС. Методы оценки. Виды оценок. Принципы формирования top 500 и других рейтингов.

10. Надежность, отказоустойчивость и другие характеристики МВС. Требования к компонентам МВС. Закон Амдала.

11. Грид-системы и облачные инфраструктуры.

12. Понятия потока и процесса. Многозадачность.

13. Библиотека POSIX Threads, функции для создания потока, завершения потока, ожидания завершения. Отсоединенные потоки (detached threads).

14. Взаимодействия потоков. Семафоры.

15. Мьютексы.

16. Условные переменные.

17. Блокировки на чтение-запись.

18. Средства многопоточного программирования в современном стандарте С++. Класс thread, классы и методы для создания и синхронизации потоков.

19. Средства синхронизации в С++ 11: mutex, lock\_guard.

20. Атомарные данные и операции.

21. Создание потока с помощью async, конструкция future.

22. Модель выполнения программы в OpenMP. Директива parallel. Общие и приватные переменные.

23. Распараллеливание циклов с помощью директивы for.

24. Директива sections.