

Спецкурс ``Сложность вычислений``

Лектор Н.К. Верещагин, 1 семестр

Что такое алгоритм и какие бывают удобные вычислительные модели? Как грамотно подсчитать, сколько времени и памяти требуется данному алгоритму при его реализации на данной вычислительной модели? Насколько отличаются время и память, необходимые реализациям данного алгоритма на различных вычислительных моделях? Почему алгоритмы не могут производить арифметические операции с действительными числами и насколько быстро они могут выполнять операции с целыми числами? Как можно доказать, что для данной задачи не существует быстрого алгоритма решения? Что такое переборные алгоритмы? Для каких задач можно достаточно точно оценить вычислительную трудность их решения? Можно ли быстро определить раскрашиваемость данного графа в 3 цвета? Бывают ли задачи, для которых доступ к датчику случайных чисел позволяет решить задачу быстрее? Как доказывалось, что одна задача сложнее другой? Что такое проблема перебора $P=?NP$? Что такое односторонние функции? Что такое надежный генератор псевдослучайных чисел и как его можно построить?

Ответы на все эти (и не только эти) вопросы будут даны в ходе спецкурса.

Прослушав спецкурс, вы узнаете:

- каковы основные вычислительные модели, используемые в оценках сложности вычислений (многоленточные машины Тьюринга, равнодоступные адресные машины, схемы из функциональных элементов, вероятностные варианты этих моделей),
- об ``эталонных`` трудных задачах.
- о популярных сложностных классах P , NP , $PSPACE$, EXP , IP и универсальных (наиболее трудных) задачах в этих классах,
- о методах установления вычислительной трудности задач различного типа (вычисления функции, поиска объекта с данными свойствами, оптимизации, аппроксимации, подсчета, обращения функций), а также научитесь: - оценивать время и память, требуемые данному алгоритму при его реализации на данной вычислительной модели,
- сводить задачи друг к другу с целью доказательства их вычислительной трудности, в частности, доказывать NP полноту и $PSPACE$ полноту задач,
- различать трудность для наихудшего входа и трудность для случайно взятого входа,
- сводить задачи друг к другу с целью доказательства их вычислительной трудности,
- строить надежные генераторы псевдослучайных чисел.

Для лучшего усвоения курса полезно знакомство с первой половиной курса ``Избранные главы дискретной математики`` (лектор И.В.Артакин).