

Модели вычислений.

Контрольные вопросы.

Раздел 1. Объекты, функции, абстракции

Тема 1. Вычисление значения.

- 1.1. Что понимается под формальной системой?
- 1.2. Как связаны выражения и означивания?
- 1.3. Приведите определение объекта.
- 1.4. Как строятся индуктивные классы?
- 1.5. В чем особенности вычисления без переменных?
- 1.6. Какова роль и место комбинаторов в вычислительных технологиях?
- 1.7. Сформулируйте назначение операции абстракции.
- 1.8. Какую роль в вычислительных технологиях играет операция применения?
- 1.9. Как выполняется операция связывания?

Тема 2. Объекты и вычисления с объектами.

- 2.1. По каким признакам отличают формальные и фактические параметры?
- 2.2. В чем заключается процесс передачи параметров?
- 2.3. Как выполняется подстановка в вычислительных технологиях?
- 2.4. В чем содержательный смысл комбинаторной характеристики?
- 2.5. Какое назначение у системы постулатов и правил вывода?
- 2.6. Охарактеризуйте отношения между объектами. Как выполняются редукция, экспансия, конверсия?
- 2.7. В чем состоит постановка задачи синтеза объекта с заданной комбинаторной характеристикой?

Тема 3. Связи между объектами.

- 3.1. Какими отображениями можно характеризовать системы объектов?
- 3.2. Для каких целей применяются неподвижные точки в вычислениях?
- 3.3. Сформулируйте теорема о неподвижной точке и обсудите ее содержательный смысл.
- 3.4. Какими способами выполняется представление циклов в

вычислениях?

3.5. Приведите определение рекурсии. Каков ее содержательный смысл?

3.6. Как строятся структуры данных? Какова их связь с системами объектов?

Раздел 2. Синтаксическая теория вычислений.

Тема 4. Системы типизации.

4.1. В чем заключается представление о типе?

4.2. Каким образом выполняется приписывание типа?

4.3. На чем основано построение содержательной интерпретации?

4.4. Какова схема построения типизированного исчисления комбинаторов?

4.5. Какова схема построения типизированного исчисления абстракций?

4.6. Какие конструкции считаются исходными типами?

4.7. Как устроены дедуктивные системы и какова схема вывода производного типа?

4.8. Каким образом строятся типы высших порядков?

4.9. Из каких компонентов строятся функциональные пространства?

Тема 5. Решение задачи синтеза структуры данных.

5.1. Как строится эквивалентная формулировка системы?

5.2. В чем содержательный смысл итеративных уточнений?

5.3. Как и зачем осуществляется синтез операторов/функций?

5.4. В чем состоит свойство полноты системы и какова практическая полезность этого свойства?

5.5. Как и зачем выполняется усиление выразительных возможностей системы?

5.6. В чем состоит формулировка и какова схема решения задачи погружения?

Тема 6. Базисы.

6.1. Как строится определение базиса?

6.2. В чем заключается свойство базисности?

6.3. Разъясните смысл фиксированных базисов, а также возможность построения нефиксированных базисов. Приведите примеры.

6.4. Какова схема решения задачи разложения объекта в базисе? В

чем заключаются границы применимости метода?

6.5. Каково назначение нумералов и комбинаторной арифметики?

Раздел 3. Динамика вычислений.

Тема 7. Динамические базисы.

7.1. Как возникает понятие о суперкомбинаторе?

7.2. В чем состоит процесс компиляции, основанный на суперкомбинаторах?

7.3. Какими способами можно выполнять сведение абстракций к суперкомбинаторам?

7.4. Из каких основных шагов состоит алгоритм подъема абстракции?

Тема 8. Использование параметров.

8.1. Какие параметры признаются избыточными? Какова этапность устранения избыточных параметров?

8.2. Зачем выполняется упорядочивание параметров? Можно ли обойтись без их упорядочивания?

Тема 9. Использование параметров (продолжение).

9.1. Какими особенностями характеризуются модифицированные алгоритмы подъема? В чем их отличие от основного алгоритма?

9.2. Как осуществляется подъем при рекурсии? Можно ли принципиально устранить рекурсию?

Тема 10. Подвыражения.

10.1. Как устанавливаются подвыражения? На каком свойстве основано выделение максимально свободных выражений?

10.2. Какие вычисления считаются жадными, а какие – ленивыми?

Как на практике используется свойство ленивости вычисления и что это дает?

Тема 11. Оптимизации.

11.1. По каким критериям выполняется оптимизация? Чем характеризуется ленивая реализация?

11.2. В чем заключается экономия в случае выполнения оптимизации?

11.3. Какие выражения предпочтительно использовать при

оптимизации?

11.4. Какие параметры и факторы используются при выполнении оптимизации?

Раздел 4. Абстрактные машины.

Тема 12. Механизмы вычислений.

12.1. В чем назначение семантики вычислений?

12.2. Какими способами осуществляются вычисления значения?

12.3. Как на практике осуществляется вызов параметра?

12.4. На каких принципах основаны системы вычисления значений?

12.5. При каких условиях встречаются коллизии переменных, почему и как этого стремятся избежать в практике вычислительных технологий?

12.6. Как выполняются вычисления в категории? Какие преимущества это обеспечивает?

Тема 13. Теории вычислений.

13.1. Какие формы принимают системы вычисления значений?

13.2. Как строятся эквациональные системы вычисления значения?

13.3. В чем смысл эквациональности? Можно ли без нее обойтись? Если “нет”, то почему? Если “да”, то каким образом?

Тема 14. Теории вычислений (продолжение).

14.1. Нужно ли оптимизировать вычисления?

14.2. По каким критериям это осуществляется?

14.3. Какие вам известны оптимизационные равенства и в чем их содержательный смысл?

14.4. Как на практике корректно осуществлять экономию в кодировании программы? Вычисления?

Тема 15. Цикл работы категориальной абстрактной машины (КАМ).

15.1. Как выделяется система инструкций КАМ?

15.2. Какую роль играет состояние вычисления? Является ли оно самой примитивной характеристикой вычисления?

15.3. Какова структура КАМ? Можно ли изменить число ее регистров?

15.4. На чем основана оптимизация исполнения кода?

Тема 16. Циклические вычисления.

- 16.1. В чем содержательный смысл среды вычислений?
- 16.2. Как организуется и выполняется на практике рекурсивная модификация среды?
- 16.3. Что считается КАМ-программой? Приведите собственные примеры КАМ-программ, способов их упрощения.

Тема 17. SAML, F# и примеры программ.

17.1. На каких принципах основана реализация языка SAML? Для решения каких классов задач из области вычислительных технологий он более всего приспособлен?

17.2. Можете ли вы указать примеры вычислений, которые нельзя реализовать на SAML?

17.3. Каковы характеристики и принципы реализации аппликативных языков?

17.4. В чем преимущества аппликативных вычислительных технологий? А недостатки?

17.5. Какую наипростейшую аппликативную программу, которая никогда не завершается, вы могли бы привести в качестве примера?

