

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления
Кафедра Интеллектуальных информационных технологий

ОТЧЁТ
по ознакомительной практике

Выполнил:

Б. А. Семченко

Студент группы
321703

Проверил:

В. Н. Тищенко

Минск 2024

СОДЕРЖАНИЕ

Введение	3
1 Постановка задачи	4
2 Формализованные фрагменты современного состояния работ в области разработки компьютеров для интеллектуальных систем . . .	6
3 Формальная семантическая спецификация библиографических источников	14
Заключение	16
Список использованных источников	16
Список использованных источников	17

ВВЕДЕНИЕ

Цель:

Закрепить практические навыки формализации информации в интеллектуальных системах с использованием семантических сетей.

Задачи:

- Построение формализованных фрагментов теории интеллектуальных компьютерных систем и технологий их разработки;
- Построение формальной семантической спецификации библиографических источников, соответствующих указанным выше фрагментам;
- Оформление конкретных предложений по развитию текущей версии Стандарта интеллектуальных компьютерных систем и технологий их разработки

1 ПОСТАНОВКА ЗАДАЧИ

Часть 6 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"

⇒ библиографическая ссылка*:

- *Голенков В.В..ТехноКПЖЦССИКСНП-2023см*
:= [Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения]
⇒ URL*:
[<https://libeldoc.bsuir.by/handle/123456789/51151>]
- *Клычева Д.М..АрхиКС-2022см*
:= [Архитектура компьютерных систем]
⇒ URL*:
[<https://cyberleninka.ru/article/n/arhitektura-kompyuternyh-setey>]
- *Ясницкий Л.Н.. ИнтеС-2016кн*
:= [Интеллектуальные системы]
⇒ URL*:
[<https://publications.hse.ru/pubs/share/folder/uxv237cikj/202053393.pdf>]
- *Грейбо С.В..АрхитВС-2019кн*
:= [Архитектура вычислительных систем]
⇒ URL*:
[<http://scipro.ru/conf/computerarchitecture.pdf>]
- *Козырева В.А.. ПринциАН-2022см*
:= [Принципы архитектуры Неймана]
⇒ URL*:
[<https://moluch.ru/archive/419/93174/>]
- *Ильасова Ф.С... ТаксоФдПВ-2018см*
:= [ТАКСОНОМИЯ ФЛИННА ДЛЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ]
⇒ URL*:
[<https://elibrary.ru/item.asp?id=36930470>]
- *Левченко Н.Н..ПаралПВСДРАуСОВСсАРМ-2008см*
:= [ПАРАЛЛЕЛЬНАЯ ПОТОКОВАЯ ВЫЧИСЛИТЕЛЬНАЯ СИСТЕМА - ДАЛЬНЕЙШЕЕ РАЗВИТИЕ АРХИТЕКТУРЫ И СТРУКТУРНОЙ ОРГАНИЗАЦИИ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ С АВТОМАТИЧЕСКИМ РАСПРЕДЕЛЕНИЕМ РЕСУРСОВ]
⇒ URL*:
[<https://elibrary.ru/item.asp?id=12160916>]
- *Черняк Л.. ЗаконАиБМП-2009см*
:= [ЗАКОН АМДАЛА И БУДУЩЕЕ МНОГОЯДЕРНЫХ ПРОЦЕССОРОВ]
⇒ URL*:
[<https://elibrary.ru/item.asp?id=12916460>]

⇒ аттестационные вопросы*:

- ⟨ • *Вопрос 1 по Части 6 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"*

- *Вопрос 2 по Части 6 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"*

}

Вопрос 1 по Части 6 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"

:= [Современное состояние работ в области разработки компьютеров для интеллектуальных систем]

⇒ библиографическая ссылка*:

- *Голенков В.В..ТехКомпПодЖЦССИКСНП-2023art*
:= [Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения]

Вопрос 2 по Части 6 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"

:= [Анализ существующих архитектур вычислительных систем]

⇒ библиографическая ссылка*:

- *Голенков В.В..ТехКомпПодЖЦССИКСНП-2023art*
:= [Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения]
- *Козырева В.А.. ПринцАН-2022ст*
:= [Принципы архитектуры Неймана]
- *Клычева Д.М..АрхиКС-2022ст*
:= [Архитектура компьютерных систем]
- *Ясницкий Л.Н.. ИнтеС-2016ст*
:= [Интеллектуальные системы]

2 ФОРМАЛИЗОВАННЫЕ ФРАГМЕНТЫ СОВРЕМЕННОГО СОСТОЯНИЯ РАБОТ В ОБЛАСТИ РАЗРАБОТКИ КОМПЬЮТЕРОВ ДЛЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

машина Фон-Неймана

- \equiv [von Neumann machine]
- \equiv [абстрактная машина фон-Неймана]
- \equiv [универсальный репликатор]
- \equiv [самовоспроизводящийся автомат]
- \equiv [абстрактная модель ЭВМ]
- \equiv [искусственная самовоспроизводящаяся система]
- \in *абстрактная машина обработки информации*
- \in *нанотехнология и молекулярная сборка*
- \in *искусственная жизнь и синтетическая биология*
- \Rightarrow *обобщённая декомпозиция**:
 - {
 - *запоминающее устройство*
 \equiv [ЗУ]
 - *арифметико-логическое устройство*
 \equiv [АЛУ]
 - *устройство управления*
 \equiv [УУ]
 - *устройства ввода и вывода*
 \equiv [ВВО]}
- \Rightarrow *разбиение**:
 - {
 - *однопроцессорная машина*
 - *многопроцессорная машина*}
- \Rightarrow *разбиение**:
 - {
 - *векторная машина*
 - *суперскалярная машина*}
- \Rightarrow *принципы, лежащие в основе**:
 - <{
 - [Информация в памяти представляется в виде последовательности строк символов в бинарном алфавите (“0” или “1”).]
 - [Память машины представляет собой последовательность адресуемых адресуемых ячеек памяти.]
 - [В каждую ячейку может быть записана любая строка символов в бинарном алфавите. При этом длина строк для всех адресуемых ячеек одинакова (в текущем стандарте ячеек, называемых байтами, равна 8 бит).]
 - [Каждой ячейке памяти взаимно однозначно соответствует битовая строка, обозначающая эту ячейку и являющаяся ее адресом.]
 - [Каждому типу элементарных действий (операций), выполняемых в памяти машины фон-Неймана, взаимно однозначно ставится ее идентификатор, который в памяти представляется также в виде битовой строки.]
 - [Программа, выполняемая в памяти, хранится в памяти в виде последовательности спецификаций конкретных операций (команд).]}

- [Таким образом, и обрабатываемые данные, и программы для обработки этих данных хранятся в одной и той же памяти (в отличие, например, от Гарвардской архитектуры) и кодируются одинаковым образом.]
- ⇒ }
автор*:
- *Голенков В.В.*
 - *Козырева В.А.*
- ⇒ библиографическая ссылка*:
- *Голенков В.В..ТехноКПЖЦССИКСНП-2023ст*
 - *Козырева В.А.. ПринцАН-2022ст*

архитектура вычислительной системы

:= [computer system architecture]

:= [ЭВМ]

⇒ разбиение*:

- {• архитектура вычислительной системы с единственной глобальной внутренней памятью
- архитектура вычислительной системы со множественной глобальной внутренней памятью

⇒ разбиение*:

- {• архитектура вычислительной системы со структурно перестраиваемыми межпроцессорными связями
- архитектура вычислительной системы без структурно перестраиваемых межпроцессорных связей

⇒ разбиение*:

- {• архитектура вычислительной системы без структурно становящейся памяти
- архитектура вычислительной системы со структурно становящейся памятью

⇒ разбиение*:

- {• архитектура вычислительной системы с ассоциативным доступом к глобальной памяти
- архитектура вычислительной системы без ассоциативного доступа к глобальной памяти

⇒ разбиение*:

- {• архитектура вычислительной системы с адресным доступом к глобальной памяти с линейным адресным пространством
- архитектура вычислительной системы без адресного доступа к глобальной памяти с линейным адресным пространством

⇒ разбиение*:

- {• архитектура вычислительной системы с системой команд регистровой обработки данных
- архитектура вычислительной системы без системы команд с регистровой обработкой данных

}

- ⇒ разбиение*:
- {• архитектура вычислительной системы с системой команд стековой обработки данных
 - архитектура вычислительной системы без системы команд стековой обработки данных
- ⇒ разбиение*:
- {• архитектура вычислительной системы с адаптивным распределением данных
 - архитектура вычислительной системы без адаптивного распределения данных
- ⇒ разбиение*:
- {• архитектура вычислительной системы исключительно с двоичным представлением данных в оперативной памяти
 - архитектура вычислительной системы не исключительно с двоичным представлением данных в оперативной памяти
- ⇒ разбиение*:
- {• архитектура вычислительной системы с исключительно дискретным представлением данных
 - архитектура вычислительной системы без исключительно дискретного представления данных
- ⇒ разбиение*:
- {• архитектура вычислительной системы с дискретным представлением данных
 - архитектура вычислительной системы без дискретного представления данных
- ⇒ автор*:
- Голенков В.В.
 - Грейбо С.В.
- ⇒ библиографическая ссылка*:
- Голенков В.В.. ТехноКПЖЦССИКСНП-2023ст
 - Грейбо С.В.. АрхитВС-2019кн

гарвардская архитектура

- := [Harvard Architectural]
- := [раздельная архитектура памяти]
- := [архитектура с независимыми памятью для инструкций и данных]
- := [двухпамятная архитектура]
- ∈ эмбедед-система
- ∈ сетевые система
- ∈ система цифровой обработки сигналов
- ⇒ отличительные признаки*:
- {• хранилище инструкций и хранилище данных представляют собой разные физические устройства
 - канал инструкций и канал данных физически разделены

⇒ *отличие от архитектуры Фон-Неймана**:

[В архитектуре фон Неймана процессор в каждый момент времени может либо читать инструкцию, либо читать/записывать единицу данных из/в памяти. Оба действия одновременно происходить не могут, поскольку инструкции и данные используют один и тот же поток (шину). В компьютере с использованием гарвардской архитектуры процессор может считывать очередную команду и оперировать памятью данных одновременно и без использования кэш-памяти. Таким образом, компьютер с гарвардской архитектурой при определенной сложности схемы быстрее, чем компьютер с архитектурой фон Неймана, поскольку потоки команд и данных расположены на отдельных физически не связанных между собой аппаратных каналах. Исходя из физического разделения шин команд и данных, разрядности этих шин могут различаться и физически не могут пересекаться.]

⇒ *обобщённая декомпозиция**:

- { • *память для инструкций*
- *память для данных*
- *центральный процессор*
- *шина данных*
- *шина адреса*
- *устройства ввода и вывода*
- := [VBO]
- }

⇒ *разбиение**:

- { • *простая гарвардская архитектура*
- ⇒ *пояснение**:

[В этой архитектуре имеются отдельные память для инструкций и данных, а также отдельные шины адреса и данных для доступа к ним. Центральный процессор выполняет инструкции из памяти для инструкций и обрабатывает данные из памяти для данных.]

- *модифицированная гарвардская архитектура*
- ⇒ *пояснение**:

[В этой вариации Гарвардской архитектуры может быть некоторое смешение памяти для инструкций и данных. Например, некоторые данные могут быть хранены в памяти для инструкций, чтобы обеспечить более эффективный доступ.]

- *гарвардская архитектура с кэш-памятью*
- ⇒ *пояснение**:

[В этой архитектуре может быть добавлена кэш-память как для инструкций, так и для данных. Кэш-память позволяет ускорить доступ к часто используемым инструкциям и данным, храня их ближе к процессору.]

- *гарвардская архитектура с приставкой Harvard-architecture von Neumann-interface*
- ⇒ *пояснение**:

[Это комбинированная архитектура, которая объединяет преимущества Гарвардской и фон-Неймановской архитектур. Возможно использование общей памяти для инструкций и данных, но с сохранением некоторых аспектов разделения данных и инструкций.]

⇒ *автор**:

Ясницкий Л.Н.
⇒ библиографический источник*:
Ясницкий Л.Н..ИнтеС-2016кн

таксономия Флинна

:= [Flynn's taxonomy]

:= [классификация потоков Флинна]

⇒ пояснение*:

[Общая классификация архитектур ЭВМ по признакам наличия параллелизма в потоках команд и данных.]

⇒ разбиение*:

{ • SISD

:= [Single Instruction Single Data]

⇒ пояснение*:

[Единственный поток команд и единственный поток данных. По сути дела это классическая машина фон Неймана. К этому классу относятся все однопроцессорные системы.]

• SIMD

:= [Single Instruction Multiple Data]

⇒ пояснение*:

[Единственный поток команд и множественный поток данных. Типичными представителями являются матричные компьютеры, в которых все процессорные элементы выполняют одну и ту же программу, применяемую к своим (различным для каждого ПЭ) локальным данным. Некоторые авторы к этому классу относят и векторно-конвейерные компьютеры, если каждый элемент вектора рассматривать как отдельный элемент потока данных.]

• MISD

:= [Multiple Instruction Single Date]

⇒ пояснение*:

[Множественный поток команд и единственный поток данных. М. Флинн не смог привести ни одного примера реально существующей системы, работающей на этом принципе. Некоторые авторы в качестве представителей такой архитектуры называют векторно-конвейерные компьютеры, однако такая точка зрения не получила широкой поддержки.]

• MIMD

:= [Multiple Instruction Multiple Date]

⇒ пояснение*:

[множественный поток команд и множественный поток данных. К этому классу относятся практически все современные многопроцессорные системы.]

}

⇒ автор*:

Ильясова Ф.С

⇒ библиографический источник*:

Ильясова Ф.С..ТаксоФдПВ-2018ст

параллельная вычислительная система

:= [parallel computing system]

∈ *вычислительная система*

∈ *компьютерная система*

⇒ *примечание**:

[Используются несколько процессоров или ядер для одновременного выполнения задач. Это позволяет разделить задачи на более мелкие части и выполнять их параллельно, что приводит к повышению производительности и сокращению времени выполнения. Параллельные вычислительные системы широко применяются в различных областях, таких как научные исследования, обработка данных, симуляции и высокопроизводительные вычисления. Однако, разработка и программирование таких систем могут представлять сложности, требуя учета синхронизации, распределения нагрузки и эффективного использования ресурсов для достижения максимальной производительности.]

⇒ *декомпозиция**:

{ • *симметрично-многопроцессорная система*

⇒ *пояснение**:

[Система, в которой несколько процессоров имеют общую память и равный доступ к всем ресурсам. Каждый процессор может выполнять независимые задачи параллельно.]

• *масштабируемая многопроцессорная система*

⇒ *пояснение**:

[Система, которая может быть масштабирована путем добавления дополнительных узлов с процессорами и памятью. Узлы могут работать параллельно, но не имеют общей памяти.]

• *кластерная система*

⇒ *пояснение**:

[Кластеры состоят из нескольких независимых компьютеров или серверов, связанных сетью. Каждый компьютер в кластере является отдельной параллельной системой, а задачи могут быть распределены между узлами кластера.]

• *массово-параллельная система*

⇒ *пояснение**:

[Системы, в которых очень большое количество процессоров или ядер работают параллельно над задачами. Примером такой системы может быть суперкомпьютер с миллионами ядер или система, основанная на графических процессорах (GPU).]

}

⇒ *автор**:

Левченко Н.Н.

⇒ *библиографический источник**:

Левченко Н.Н..ПаралПВСДРАиСОВСсАРМ-2008ст

закон Амдала

:= [Amdahl's law]

:= [закон Амдала об ускорении]

∈ *оптимизация производительности*

:= [Speedup=1/((1-f))+f/n]

⇒ *разбиение**:

{ • *Speedup*

⇒ *пояснение**:

- f
 \Rightarrow *пояснение**:
 [Относительное ускорение.]
 - n
 \Rightarrow *пояснение**:
 [Часть кода, которая может быть распараллелена.]
 - n
 \Rightarrow *пояснение**:
 [Число параллельных процессоров.]
- }
 \Rightarrow *автор**:
 Черняк Л.
 \Rightarrow *библиографический источник**:
 Черняк Л.. Закон АиБМП-2009 ст

облачное вычисление

- $:=$ [cloud computing]
 $:=$ [модель обеспечения удобного сетевого доступа по требованию к некоторому общему фонду]
 \Rightarrow *характеристики**:
- { • *самообслуживание по требованию*
 \Rightarrow *пояснение**:
 [Потребитель самостоятельно определяет свои вычислительные потребности: серверное время, скорости доступа и обработки данных, объём хранимых данных — без взаимодействия с представителем поставщика услуг.]
 - *универсальный доступ по сети*
 \Rightarrow *пояснение**:
 [Услуги доступны потребителям по сети передачи данных вне зависимости от используемого терминального устройства.]
 - *объединение ресурсов*
 \Rightarrow *пояснение**:
 [Поставщик услуг объединяет ресурсы для обслуживания большого числа потребителей в единый пул для динамического перераспределения мощностей между потребителями в условиях постоянного изменения спроса на мощности; при этом потребители управляют только основными параметрами услуги (например, объёмом данных, скоростью доступа), но фактическое распределение ресурсов, предоставляемых потребителю, осуществляет поставщик (в некоторых случаях потребители всё-таки могут управлять некоторыми физическими параметрами перераспределения, например, указывать желаемый центр обработки данных из соображений географической близости).]
 - *эластичность*
 \Rightarrow *пояснение**:
 [Услуги могут быть предоставлены, расширены, сужены в любой момент времени, без дополнительных издержек на взаимодействие с поставщиком, как правило, в автоматическом режиме.]
- }
 \Rightarrow *разбиение**:
 { • *публичное облако*

- *частное облако*
- *гибридное облако*

}

⇒ *автор*:*

Устинов В.А.

⇒ *библиографический источник*:*

Устинов В.А..ВведевОВ-2016ст

3 ФОРМАЛЬНАЯ СЕМАНТИЧЕСКАЯ СПЕЦИФИКАЦИЯ БИБЛИОГРАФИЧЕСКИХ ИСТОЧНИКОВ

Голенков В.В..ТехноКПЖЦССИКСНП-2023ст

⇒ *ключевой знак**:

- *машина Фон-Неймана*
- *архитектура вычислительных систем*

⇒ *аннотация**:

[В издании представлено описание текущей версии открытой технологии онтологического проектирования, производства и эксплуатации семантически совместимых гибридных интеллектуальных компьютерных систем (Технологии OSTIS). Предложена стандартизация интеллектуальных компьютерных систем, а также стандартизация методов и средств их проектирования, что является важнейшим фактором, обеспечивающим семантическую совместимость интеллектуальных компьютерных систем и их компонентов, что существенно снижает трудоемкость разработки таких систем. Книга предназначена всем, кто интересуется проблемами искусственного интеллекта, а также специалистам в области интеллектуальных компьютерных систем и инженерии знаний. Может быть использована студентами, магистрантами и аспирантами специальности «Искусственный интеллект». Табл. 8. Ил. 223. Библиогр.: 665 назв.]

⇒ *URL**:

[<https://libeldoc.bsuir.by/handle/123456789/51151>]

Клычева Д.М..АрхиКС-2022ст

⇒ *аннотация**:

[В данной статье рассматриваются особенности создания цифровых технологий и их использование в развитии. Приведены методы и стратегии влияния системы развития технологий в период цифровизации всех отраслей. Даны рекомендации по внедрению технологий в отрасль.]

⇒ *цитата**:

[Планирование сетевой архитектуры имеет жизненно важное значение, поскольку оно либо повышает, либо снижает производительность всей системы.]

⇒ *URL**:

[<https://cyberleninka.ru/article/n/arhitektura-kompyuternyh-setey>]

Ясницкий Л.Н..ИнтеС-2016кн

⇒ *ключевой знак**:

гарвардская архитектура

⇒ *цитата**:

[Человек — это самый сложный из доступных для нашего восприятия объект, а способность мышления — его главное свойство — атрибут. Искусственный интеллект — это наука, целью которой является изучение и моделирование атрибута человека — мышления.]

⇒ *URL**:

[<https://publications.hse.ru/pubs/share/folder/uxv237cikj/202053393.pdf>]

Грейбо С.В..АрхитВС-2019кн

⇒ *ключевой знак**:

- ⇒ *архитектура вычислительной системы*
*цитата**:
[Под архитектурой ЭВМ принято понимать совокупность общих принципов организации аппаратно-программных средств и основных их характеристик, определяющая функциональные возможности ЭВМ при решении соответствующих типов задач.]
- ⇒ *URL**:
[<http://scipro.ru/conf/computerarchitecture.pdf>]

Козырева В.А.. ПринциАН-2022ст

- ⇒ *ключевой знак**:
Машина Фон-Неймана
- ⇒ *аннотация**:
[В статье авторы рассказывают про архитектуры ПК и про принципы Джона фон Неймана, а также приводят направления критики его принципов.]
- ⇒ *URL**:
[<https://moluch.ru/archive/419/93174/>]

Ильясова Ф.С..ТаксоФдПВ-2018ст

- ⇒ *ключевой знак**:
таксономия Флинна
- ⇒ *аннотация**:
[В статье представлен анализ таксономии Флинна для параллельных вычислений в контексте построения виртуальной лаборатории. Также проанализированы работы известных авторов в области параллельных вычислений. Представлен пример организации виртуальной лаборатории на базе Крымского инженерно-педагогического университета, как способ организации взаимодействия субъектов «Преподаватель-Студент».]
- ⇒ *URL**:
[<https://elibrary.ru/item.asp?id=36930470>]

Левченко Н.Н..ПаралПВСДРАиСОВСсАРМ-2008ст

- ⇒ *ключевой знак**:
параллельная вычислительная система
- ⇒ *аннотация**:
[Рассматривается базовая архитектура и структурная организация параллельной потоковой вычислительной системы (ППВС). Оцениваются достоинства и преимущества использования нетрадиционной модели вычислений с управлением потоком данных, аппаратной ее реализации, описываются общие принципы функционирования ППВС. Предлагаются варианты аппаратной реализации системы в виде многоядерного кристалла, а также пути развития и масштабирования данной системы.]
- ⇒ *URL**:
[<https://elibrary.ru/item.asp?id=12160916>]

ЗАКЛЮЧЕНИЕ

В рамках учебно-ознакомительной практики были получены знания в предметной области "Современное состояние компьютеров для интеллектуальных систем". Повышены навыки в формализации понятий на `scn-latex`. А также улучшены навыки в фильтрации и поиске информации.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] В.В.Голенков,. Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения / В.В.Голенков. — Бестпринт, 2023. — С. 1037.

[2] Грейбо, С.В. Архитектутра вычислительных систем / С.В. Грейбо. — 2019. — С. 77.

[3] Д.М.Клычев,. Архитектура компьютерных систем / Д.М.Клычев. — Вестник науки, 2022. — С. 20.

[4] Колесников, А.В. Гибридные интеллектуальные системы. Теория и технология разработки / А.В. Колесников. — под ред. А.М. Яшина, 2008. — С. 711.

[5] Ясницкий, Л.Н. Интеллектуальные системы / Л.Н. Ясницкий. — Лаборатория знаний, 2016. — С. 222.