

2 Выбор и обоснование средств проектирования.

2.1 Описание критериев и альтернатив

Самыми важными требованиями к ПО для автоматической авторизации пользователей являются переносимость и скорость работы. Поэтому можно выделить следующие языки программирования: C++, C#, Python, Delphi, Visual Basic.

Для выбора ЯП применим метод анализа иерархий. Выберем следующие критерии:

- а) Скорость работы — поскольку распознавание ресурсоемкий процесс, то скорость работы конечного продукта является важной характеристикой;
- б) Удобство использования — характеристика показывающая удобство использования альтернативы в проектах такого рода;
- в) Скорость разработки — характеристика показывающая продолжительность разработки ПО и простоту написания программ;
- г) Дружелюбность сообщества — легко ли получить помощь от сообщества пользователей данного ПО, в случае непредвиденной ситуации;
- д) Серьезность поддержки — на сколько авторы данного ПО серьезно относятся к своему продукту;
- е) Возможности языка - возможности предоставляемые языком для решения разнообразных задач;

					Выбор и обоснование средств проектирования			
Изм	Лист	№ докум.	Подпись	Дата				
Разраб.					ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ АВТОМАТИЧЕСКОЙ АВТОРИЗАЦИИ ПОЛЬЗОВАТЕЛЕЙ ОС UNIX	Лит.	Лист	Листов
Руковод.								
Консул.						СКФ БГТУ им. В.Г.Шухова, ПВ-41		
Н. Контр.								
Зав.каф.	Поляков В.М.							

ж) Простота компиляции и установки конечного ПО — к данной характеристике относится простота и скорость компиляции, простота установки конечного продукта;

з) Кроссплатформенность — возможность компиляции и создания установочных пакетов на различные операционные системы и возможность компиляции для различных архитектур процессоров.

Критерии выбраны исходя из личных предпочтений автора проекта и условий, в которых будет функционировать разрабатываемый продукт.

2.2 Метод анализа иерархий (МАИ)

2.2.1 Общие сведения

СППР, основанная на методе анализа иерархий (МАИ), является простым и удобным средством, которое поможет структурировать проблему, построить набор альтернатив, выделить характеризующие их факторы, задать значимость этих факторов, оценить альтернативы по каждому из факторов, найти неточности и противоречия в суждениях ЛПР/эксперта, проранжировать альтернативы, провести анализ решения и обосновать полученные результаты.

СППР МАИ может использоваться при решении следующих типовых задач:

- оценка качества организационных, проектных и конструкторских решений;
- определение политики инвестиций в различных областях;
- задачи размещения (выбор места расположения вредных и опасных производств, пунктов обслуживания);
- распределение ресурсов;
- проведение анализа проблемы по методу “стоимость-эффективность”;

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

- стратегическое планирование;
- проектирование и выбор оборудования, товаров;
- выбор профессии, места работы, подбор кадров.

Основные положения метода анализа иерархий были разработаны известным американским математиком Т.Л.Саати и опубликованы в 1977г [1]. Томас Саати является одним из самых ярких представителей прикладной науки. Об этом говорят не только его математическая эрудиция и глубина новых теоретических результатов, но и диапазон приложений. Он был прав, предпослав к одной из своих монографий эпиграф: «Я люблю обе стороны математики: чистую — как возвышенный уход от реальности, прикладную - как страстное стремление к жизни».

МАИ используется для решения слабо структурированных и не структурированных проблем. Методология решения таких проблем опирается на системный подход, при котором проблема рассматривается как результат взаимодействия и, более того, взаимозависимости множества разнородных объектов, а не просто как их изолированная и автономная совокупность.

Человеку присущи два характерных признака аналитического мышления: один - умение наблюдать и анализировать наблюдения, другой – способность устанавливать отношения между наблюдениями, оценивая уровень (интенсивность) взаимосвязей, а затем синтезировать эти отношения в общее восприятие наблюдаемого.

На основе этих свойств человеческого мышления были сформулированы три принципа, реализация которых и является содержанием МАИ:

- принцип идентичности и декомпозиции;
- принцип дискриминации и сравнительных суждений;
- принцип синтеза.

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

2.2.2 Принцип идентичности и декомпозиции

Реализация этого принципа осуществляется на первом этапе применения МАИ, в котором предусматривается структурирование проблемы в виде иерархии. Иерархия строится с вершины - это общая цель или фокус проблемы. В общем случае целей может быть несколько. За фокусом следует уровень наиболее важных критериев. Каждый из критериев может разделяться на субкритерии, за которыми следует уровень альтернатив. ЛПР при построении иерархии вынужден вникнуть в проблему. От этого этапа во многом зависят конечные результаты принятия решений. Формирование множества альтернатив и критериев осуществляется с учетом рекомендаций. Этап является неформализуемым.

Пример. При обсуждении проблемы улучшения жилищных условий семьей была сформулирована цель - покупка дома. Обсуждались и другие цели решения этой проблемы (например, ремонт имеющегося жилья). Из каталога были отобраны три наиболее предпочтительных дома (варианты А, В, С), которые и были осмотрены семьей непосредственно. Для выбора окончательного варианта она решила воспользоваться методом анализа иерархий. Итогом первого этапа МАИ, который явился результатом семейного обсуждения, стала следующая иерархия (рисунок 2.1): иерархия - есть определенный тип системы, основанный на предположении, что элементы системы могут группироваться в несвязанные множества. Элементы каждой группы находятся под влиянием элементов другой группы и в свою очередь оказывают влияние на элементы следующей группы.

Считается, что элементы в каждой группе иерархии (называемые уровнем, кластером, стратой) независимые (рисунок 2.2).

Математически иерархия и ее свойства могут быть описаны следующим образом. На множестве объектов $I = \{1, 2, \dots, N\}$ определяется иерархическая структура путем задания орграфа $G = (I, W) W \subset I \times I$, который:

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

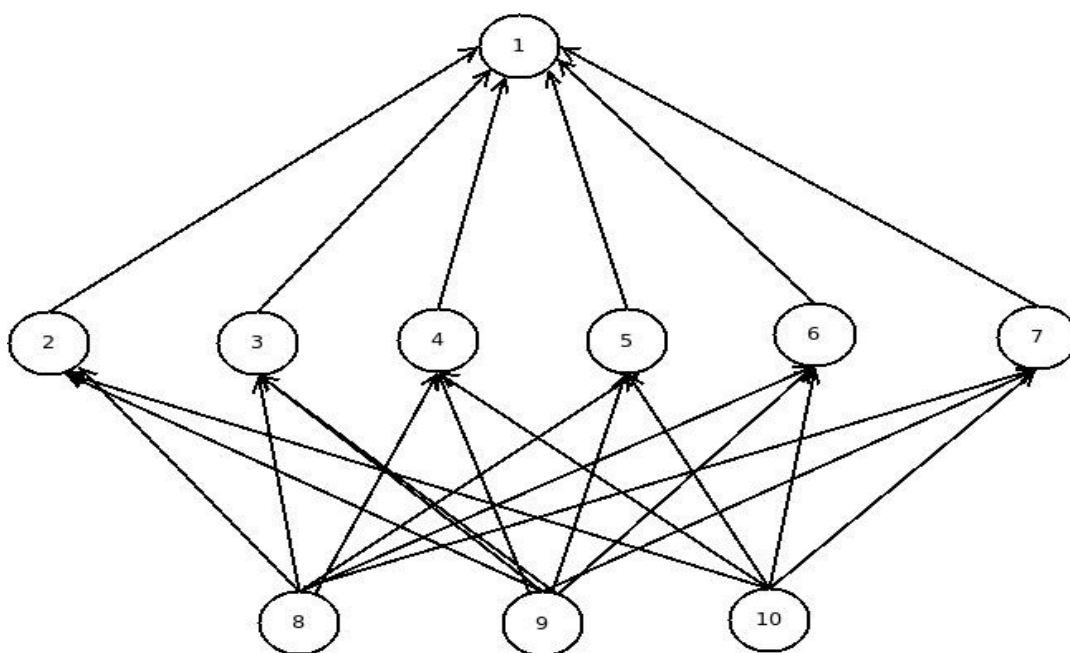


Рисунок 2.1 — Иерархия проблемы улучшения жилищных условий.

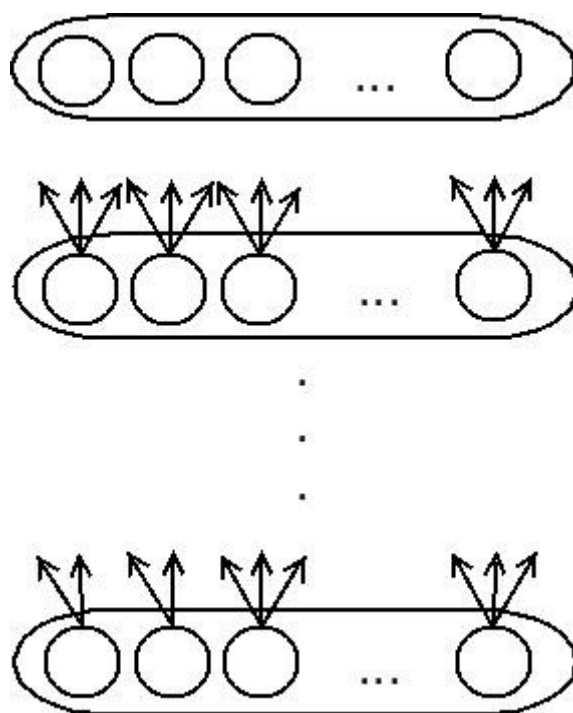


Рисунок 2.2 — Общий вид иерархии

а) разбивает вершины на не пересекающиеся уровни:

$$I = \bigcup_{l=1}^m V_l; l = \overline{1, m}; V_i \cap V_j = \emptyset; i, j = \overline{1, m};$$

б) $(i, j) \in W$ означает, что вес Z_i объекта i непосредственно зависит от веса Z_j объекта j ;

в) если (i, j) - дуга графа G , т.е. $(i, j) \in W$, то объекты i и j находятся на смежных уровнях, т.е. найдется такое k , что $i \in V_{k+1}$, $j \in V_k$;

г) веса Z_i объекта $i \in V_{k+1}$ определяются через веса Z_j вершин множества $L_i = \{j \mid (i, j) \in W\} \subseteq V_k$, в которые ведут дуги из вершины i с помощью феноменологически вводимой зависимости

$$Z_i = \sum_{j \in L_i} \mathfrak{J}_{ij} Z_j, i \in I \setminus V_1$$
, где \mathfrak{J}_{ij} - вес дуги (i, j) . Методика определения \mathfrak{J}_{ij} , заключается использовании знаний ЛПР, для заполнения \mathfrak{J}_{ij} дуг $(i, j) \in W$ и веса Z объектов первого уровня ($j \in V_1$).

2.2.3 Принцип дискриминации и сравнительных суждений

Данный принцип реализуется на втором этапе МАИ. Суть его заключается в том, что, используя суждения ЛПР/эксперта и определенные алгоритмы их обработки, устанавливаются веса \mathfrak{J}_{ij} дуг $(i, j) \in W$ и веса Z_j объектов первого уровня ($j \in V_1$). Если на первом уровне один объект, то вес его принимается за 1 ($Z_1=1$).

Суждения ЛПР/эксперта являются результатом исследования его структуры предпочтений. При этом исследовании применяется метод парных сравнений, содержание которого состоит в следующем. Пусть задано некоторое фиксированное множество объектов $K = \{k_i\}, i = \overline{1, n}, K \subset I$, которые сравниваются попарно с точки зрения их предпочтительности, желательности, важности и т.п. Результаты записываются в виде матрицы парных сравнений $R = \{R_{ij}\}, i, j = \overline{1, n}$.

Результат сравнения отражает не только факт, но и степень (силу, интенсивность и т.п.) превосходства. При этом используется шкала

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

относительной важности, выбор которой зависит от следующих требований:

- шкала должна давать возможность улавливать различия в ощущениях людей, когда они проводят сравнение;
- диапазон измеряемой интенсивности шкалы должен соответствовать результатам когнитивной психологии.

Удовлетворяет этим требованиям шкала, приведенная в таблице 2.1.

Из шкалы следует свойство гомогенности (однородности) объектов. Это свойство соответствует способности людей сравнивать объекты, которые не слишком сильно отличаются друг от друга. Гомогенность существенна для сравнения объектов одного порядка, т.к. человеческий разум склонен к допущению больших ошибок при сравнении не сопоставимых элементов. Когда эта несопоставимость большая, объекты располагаются в отдельные кластеры сравниваемых размеров, что выдвигает идею об уровнях и их декомпозиции.

Пример. Рассмотрим метод парных сравнений на примере покупки дома (рисунок 2.3).

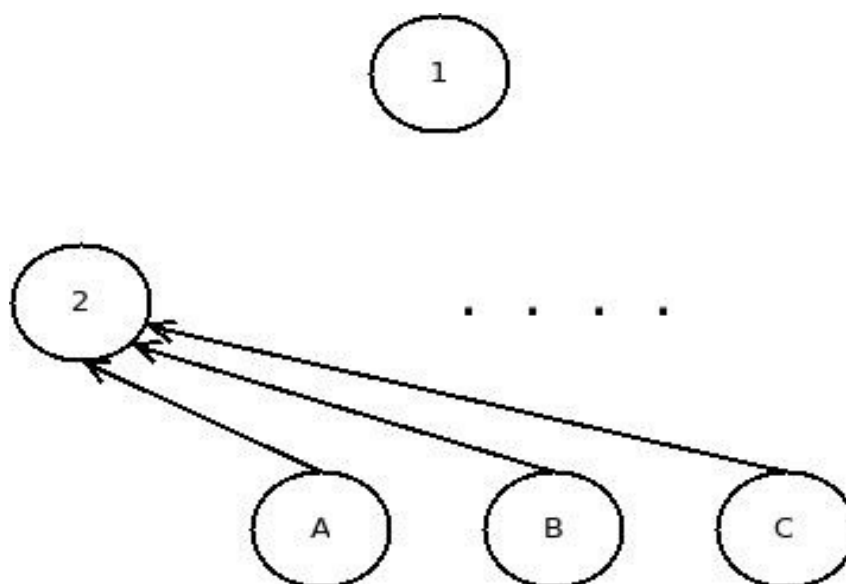


Рисунок 2.3 — Иллюстрация к методу парных сравнений.

Таблица 2.1 — Шкала относительной важности

Количественная интенсивности относительной важности	оценка	Качественная интенсивности относительной важности	оценка	Пояснения
1		Равная важность		Равный вклад двух объектов
3		Средняя превосходства одного над другим	степень	Опыт и суждения дают легкое превосходство одного объекта над другим
5		Умеренно сильное		Опыт эксперта позволяет считать одну из альтернатив явно предпочтительнее другой
7		Значительное превосходство		Один объект имеет настолько сильное превосходство, что оно становится практически значительным
9		Очень сильное превосходство		Очевидность превосходства одного объекта над другим подтверждается наиболее сильно
2,4,6,8		Промежуточные решения между двумя соседними суждениями		Применяются в компромиссном случае
Обратные приведенных выше чисел	величины	Если объекту i при сравнении с объектом j приписывается одно из приведенных выше чисел, то действию j при сравнению с i приписывается обратное значение		Применяются для сравнения обратных значений

Допустим необходимо оценить предпочтения ЛПР/эксперта на множестве вариантов А,В,С относительно критерия - размера дома. Лучше всего эту задачу

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

свести к заполнению таблицы 2.2.

Таблица 2.2 — Матрица парных сравнений к рисунку 2.3.

Размер дома	Вариант А	Вариант В	Вариант С
Вариант А	1	1/3	5
Вариант В	3	1	1/7
Вариант С	1/5	7	1

Размерность таблицы определяется количеством дуг, которые входят в рассматриваемую вершину. Элементы таблицы $r_{ij}, i, j = \overline{1,3}$ являются количественной оценкой интенсивности предпочтения i -го объекта, находящегося в i -й строке, относительно j -го объекта, находящегося в j -м столбце, в соответствии с выше рассмотренной шкалой.

При этом сравнении ЛПР/эксперту задавался следующий вопрос : насколько один вариант (например А) превосходит по размеру другой вариант (например С)? Ответом ЛПР/эксперта, как следует из таблицы, было следующее суждение: существенное или сильное превосходство.

Таким же образом осуществляется оценка предпочтений ЛПР/эксперта относительно остальных критериев путем заполнения еще пяти аналогичных матриц размерностью 3×3 . После чего метод парных сравнений распространяется на множество самих критериев относительно цели - покупки дома. В этом случае ЛПР/эксперту задается следующий вопрос: насколько важнее один критерий (например, размер дома) для реализации цели по сравнению с другим (например, финансовые условия)? Как следует из иерархии, размерность этой таблицы 6×6 .

Принимая во внимание свойство матрицы, т.е. $\forall i, j = \overline{1, n}, r_{ij} = \frac{1}{r_{ji}}$ и, как следствие, $r_{ij} = 1$ количество вопросов равно $\frac{n*(n-1)}{2}$.

Формализацией понятия непротиворечивости для метода парных сравнений является выполнение следующего равенства:

$$r_{ij}^* = r_{ik}^* * r_{kj}^*, \forall i, j, k \quad (1)$$

где r_{ij}^* - это элементы матрицы R^* , полученные в результате идеально согласованного эксперимента.

Соотношение (1) соответствует правилу логического вывода, которое в этом случае формулируется следующим образом: если i -й объект предпочтительнее k -го объекта на r_{ik}^* и k -й объект предпочтительнее j -го объекта на r_{kj}^* , то i -й объект предпочтительней j -го объекта на r_{ij}^* , причем $r_{ij}^* = r_{ik}^* * r_{kj}^*$.

Если матрица R^* обладает свойством (1), то тогда существуют такие числа $\mathfrak{Z}_i^* > 0$, что имеет место равенство:

$$r_{ij}^* = \frac{\mathfrak{Z}_i^*}{\mathfrak{Z}_j^*}, \forall i, j = \overline{1, n} \quad (2)$$

Числа $\mathfrak{Z}_i, i = \overline{1, n}$ отождествляются с весами дуг (это множество W в графе G) либо с весами объектов первого уровня (это $Z_i, i \in V_1$).

Матрица R^* имеет единичный ранг, $\{\mathfrak{Z}_i^*\}, i = \overline{1, n}$ - собственный вектор матрицы, где n - соответствующее ей собственное число.

Действительно,

$$\begin{pmatrix} \frac{\mathfrak{Z}_1^*}{\mathfrak{Z}_1^*} & \frac{\mathfrak{Z}_1^*}{\mathfrak{Z}_2^*} & \dots & \frac{\mathfrak{Z}_1^*}{\mathfrak{Z}_n^*} \\ \frac{\mathfrak{Z}_2^*}{\mathfrak{Z}_1^*} & \frac{\mathfrak{Z}_2^*}{\mathfrak{Z}_2^*} & \dots & \frac{\mathfrak{Z}_2^*}{\mathfrak{Z}_n^*} \\ \frac{\mathfrak{Z}_3^*}{\mathfrak{Z}_1^*} & \frac{\mathfrak{Z}_3^*}{\mathfrak{Z}_2^*} & \dots & \frac{\mathfrak{Z}_3^*}{\mathfrak{Z}_n^*} \\ \dots & \dots & \dots & \dots \\ \frac{\mathfrak{Z}_n^*}{\mathfrak{Z}_1^*} & \frac{\mathfrak{Z}_n^*}{\mathfrak{Z}_2^*} & \dots & \frac{\mathfrak{Z}_n^*}{\mathfrak{Z}_n^*} \end{pmatrix} * \begin{pmatrix} \mathfrak{Z}_1^* \\ \mathfrak{Z}_2^* \\ \dots \\ \mathfrak{Z}_n^* \end{pmatrix} = \begin{pmatrix} n * \mathfrak{Z}_1^* \\ n * \mathfrak{Z}_2^* \\ \dots \\ n * \mathfrak{Z}_n^* \end{pmatrix} = n * \begin{pmatrix} \mathfrak{Z}_1^* \\ \mathfrak{Z}_2^* \\ \dots \\ \mathfrak{Z}_n^* \end{pmatrix} \quad \text{или} \quad R^* \overline{\mathfrak{Z}^*} = n \overline{\mathfrak{Z}^*} \quad (3)$$

Практически добиться полной согласованности (т.е. непротиворечивости)

						Лист
					Выбор и обоснование средств проектирования	
Изм.	Лист	№ докум.	Подпись	Дата		

суждений ЛПР/эксперта далеко не всегда возможно.

Поэтому в общем случае r_{ij} будут отклоняться от “идеальных” $r_{ij}^* = \frac{z_i^*}{z_j^*}$,

вследствие чего соотношения 1, 2, 3 не будут иметь место.

Для дальнейшего анализа полезными являются следующие два факта из теории матриц:

Во-первых, если $\lambda_1, \dots, \lambda_n$ являются собственными числами матрицы R и если $r_{ij}=1, i=\overline{1, n}$, то $\sum_{i=1}^n \lambda_i = n$. Согласно этому утверждению, если имеет место (3) (т.е. матрица является идеально согласованной), то все собственные числа ее - нули, за исключением одного, равного n.

Во-вторых, если элемент положительной обратно-симметричной матрицы R незначительно изменить, то собственные числа этой матрицы также изменятся незначительно, т.е. они являются непрерывными функциями ее элементов.

Объединяя эти результаты, находим, что при малых изменениях r_{ij} от r_{ij}^* наибольшее собственное число λ_{max} (практически получаемой матрицы R при использовании метода парных сравнений) остается близким к n, а остальные собственные значения - близкими к нулю.

Отсюда можно сформулировать следующую задачу: для нахождения весов дуг или объектов первого уровня по полученной в результате метода парных сравнений матрице R необходимо определить собственный вектор \bar{J} , соответствующий максимальному собственному числу, т.е. решить уравнение:

$$R \bar{J} = \lambda_{max} \bar{J} \quad (4)$$

Так как малые изменения в $r_{ij}, i, j = \overline{1, n}$ вызывают малое изменение λ_{max} , отклонение последнего от n является мерой согласованности. Она может быть выражена с помощью индекса согласованности (ИС):

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

$$ИС = \frac{\lambda_{max} - n}{n - 1} \quad (5)$$

Если $ИС \leq 0.1$, то практически считается, что мера согласованности находится на приемлемом уровне. Индекс согласованности матрицы парных сравнений, элементы которой сгенерированы случайным образом, называется случайным индексом (СИ). Ниже представлена таблица соответствия порядка и среднего значения СИ, определенная на базе 100 случайных выборок (табл. 2.3) [1]. Отношение ИС к среднему СИ для матрицы того же порядка называется отношением согласованности (ОС). Значение ОС меньшее или равное 0,10 считается приемлемым. Обычно ИС и ОС указываются в процентах. Согласно определению, ИС можно трактовать как отклонение от идеально проведенного эксперимента (метода парных сравнений), а ОС указывает, на сколько оцениваемая степень согласованности сходится со степенью согласованности самого неидеально проведенного эксперимента.

Таким образом, МАИ допускает несогласованность (как неотъемлемую часть метода), признавая, что человеческие суждения находятся в постоянном процессе изменения и эволюции (поэтому не следует настаивать на 100% согласованности, так как суждения могут измениться после того, как проблема решена). Но надежные решения не могут быть приняты без приемлемого уровня согласованности.

Таблица 2.3 — Таблица средних значений СИ

Порядок матрицы	1	2	3	4	5	6	7	8	9	10
СИ	0,00	0,00	0,58	0,9	1,12	1,24	1,32	1,41	1,45	1,49

Для нахождения максимального собственного числа и соответствующего ему собственного вектора используется так называемый степенной метод, основанный на итерационном алгоритме

$$\lambda_{max} \approx \frac{\mathfrak{Z}_i^{(m+1)}}{\mathfrak{Z}_i^{(m)}} \quad (6)$$

где $\mathfrak{Z}_i^{(m)}$ - i-ая координата вектора $\mathfrak{Z}_i^{(m)}$; m — номер итерации.

Если принять достаточно большой номер итерации m, то можно с любой точностью получить λ_{max} . Для нахождения λ_{max} можно использовать любую координату вектора m и, в частности, можно взять среднее арифметическое:

$$\lambda_{max} = \frac{1}{n} \sum_{i=1}^n \frac{\mathfrak{Z}_{i(m+1)}}{\mathfrak{Z}_{i(m)}} \quad (7)$$

В качестве собственного вектора матрицы, соответствующего λ_{max} , принимается нормированный вектор, вычисленный по рекуррентному выражению:

$$\overline{\mathfrak{Z}^{m+1}} = R \overline{\mathfrak{Z}^m} \quad (8)$$

При $m=0$ $\overline{\mathfrak{Z}^{(0)}}$ - произвольный начальный вектор.

2.2.4 Принцип синтеза

Реализация принципа синтеза составляет содержание третьего этапа. Искомые веса объектов определяются последовательно, начиная со второго уровня иерархии (рисунок 2.4) в соответствии с решающим правилом:

$$Z_i = \sum_{j \in L_i} \mathfrak{Z}_{ij} Z_j, \forall i \in V_2, \dots, i \in V_m \quad (9)$$

Веса объектов, принадлежащих уровню альтернатив, можно считать как результат измерения их в шкале отношений в диапазоне [0,1].

Согласованность всей иерархии C определяется по следующему выражению:

$$C = \frac{\sum_{i \in D} IC_i Z_i}{\sum_{i \in D} CI_i Z_i} \quad (10)$$

где $D = IV_m$; IC_i, CH_i - соответственно индекс согласованности и случайный индекс таблицы парных сравнений, рассмотренной относительно i -го объекта.

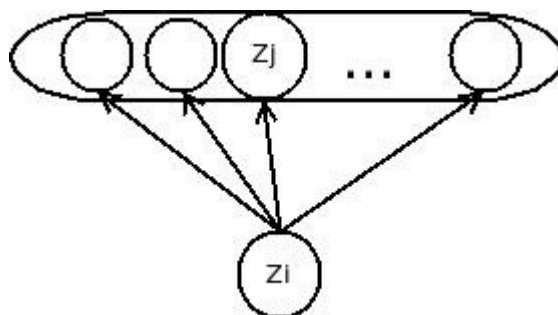


Рисунок 2.4 – Фрагмент иерархии

Приемлемым является значение C меньше или равное 10%. В противном случае качество суждений следует улучшить. Возможно, следует пересмотреть формулировку вопросов при проведении парных сравнений. Если это не поможет улучшить согласованность, то, вероятно, задачу следует более точно структурировать, т.е. вернуться к этапу 1.

2.2.5 Общая оценка МАИ как метода принятия решений

Принятие решений складывается в междисциплинарную область исследований, в которой работают психологи, математики, программисты, экономисты, инженеры. Отметим, что эта междисциплинарность является как бы переходным этапом к появлению новой дисциплины, в рамках которой специалисты будут обладать необходимыми научными знаниями из приведенных выше дисциплин, а также новыми знаниями по проблемам, ранее не изучавшимся.

Рассмотрим, насколько удовлетворяет МАИ ряду требований к научному обоснованию методов принятия решений:

- а) В МАИ способы получения информации от ЛПР/эксперта соответствуют данным когнитивной психологии о возможностях человека

перерабатывать информацию. Действительно, гомогенность и принцип иерархической декомпозиции приводят в соответствие проблему получения оценок с психометрическими возможностями человека;

б) В МАИ имеется возможность проверки информации, полученной от ЛПР/эксперта на непротиворечивость, посредством индекса и отношения согласованности как для отдельных матриц, так и для всей иерархии;

в) Любые соотношения между вариантами решений в МАИ объяснимы на основе информации, полученной от ЛПР/экспертов. Так, анализ весов объектов по нисходящим уровням иерархии позволяет понять, как получено то или иное значение веса;

г) Математическая правомочность решающего правила в МАИ прозрачна и базируется на методе собственных значений и принципе иерархической композиции, имеющих четкое математическое обоснование. Таким образом, МАИ удовлетворяет четырем основным критериям, обеспечивающим всестороннюю научную обоснованность метода принятия решений.

2.3 Выбор с использованием МАИ

2.3.1 Матрицы парных сравнений

В результате получилась иерархия, представленная на рисунке 2.5. Суждения ЛПР представлены в таблицах [2.4-2.12].

2.3.2 Результат сравнения

В результате альтернативы получили следующие оценки:

а) C++ - 0,32;

б) C# - 0,21;

в) Python — 0,23;

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

г) Delphi — 0,16;

д) Visual Basic — 0,15.

По результатам МАИ был выбран язык C++.

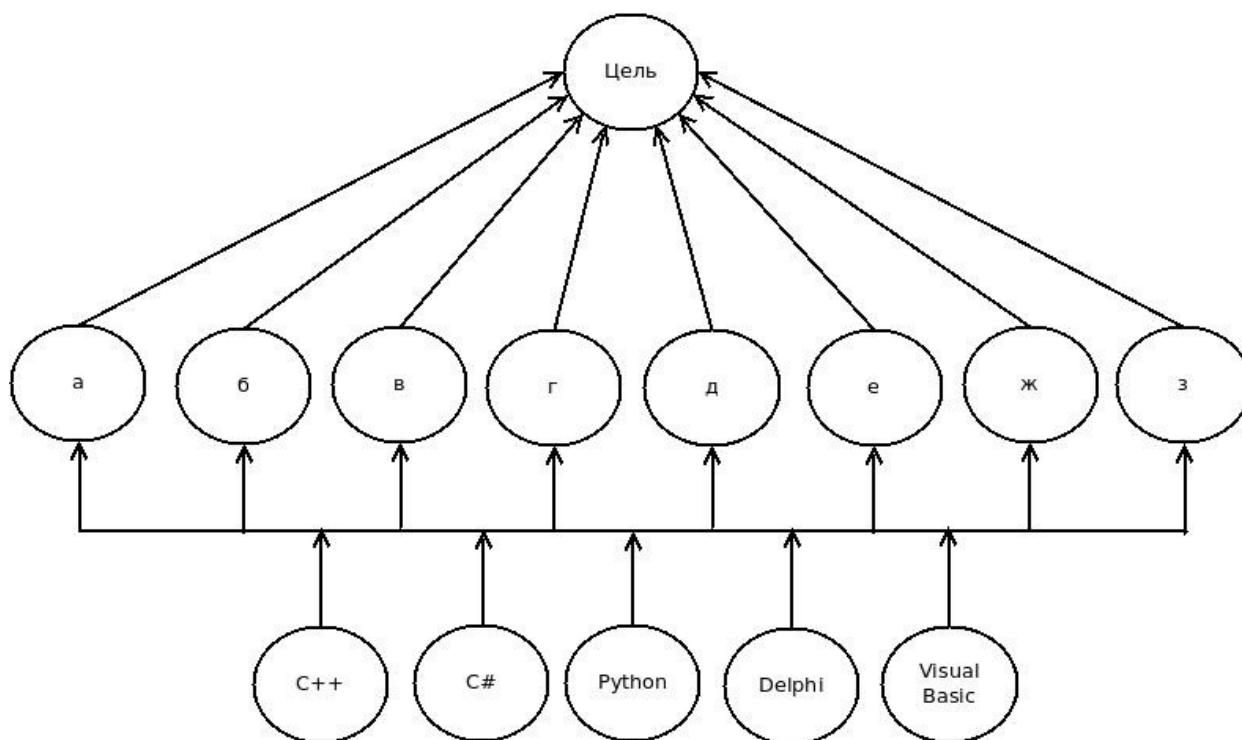


Рисунок 2.5 — Иерархия проблемы

C++ — компилируемый статически типизированный язык программирования общего назначения. Поддерживая разные парадигмы программирования, сочетает свойства как высокоуровневых, так и низкоуровневых языков[3]. В сравнении с его предшественником — языком C, — наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования[2]. Название «C++» происходит от языка C, в котором унарный оператор ++ обозначает инкремент переменной.

Являясь одним из самых популярных языков программирования[4][5], C++ широко используется для разработки программного обеспечения. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых

Таблица 2.4 — Веса критериев относительно цели

Выбор ЯП	Скорость работы	Удобство использования	Скорость разработки	Дружелюбность сообщества	Серьезность поддержки	Возможности ЯП	Простота компиляции и установки	Кросс-платформенность	Веса
Скорость работы	1	1/7	1/5	1/7	1/7	1/5	1/3	1/2	0.02
Удобство использования	7	1	5	1/3	1/3	3	3	5	0.16
Скорость разработки	5	1/5	1	1/5	1/5	2	2	3	0.08
Дружелюбность сообщества	7	3	5	1	1/2	5	5	7	0.26
Серьезность поддержки	7	3	5	2	1	5	5	7	0.32
Возможности ЯП	5	1/3	1/2	1/5	1/5	1	2	5	0.07
Простота компиляции и установки	3	1/3	1/2	1/5	1/5	1/2	1	5	0.06
Кросс-платформенность	2	1/5	1/3	1/7	1/7	1/5	1/5	1	0.03
ИС = 9.9%					ОС = 7%				

Таблица 2.5 — Веса альтернатив относительно критерия «Скорость работы»

Скорость работы	C++	C#	Python	Delphi	Visual Basic	Веса
C++	1	1/4	1/5	1/7	1/7	0.04
C#	4	1	1/3	1/5	1/5	0.08
Python	5	3	1	1/3	1/3	0.16
Delphi	7	5	3	1	3	0.44
Visual Basic	7	5	3	1/3	1	0.28
ИС = 9.2%			ОС = 8.2%			

Таблица 2.6 — Веса альтернатив относительно критерия «Удобство использования»

Удобс тво использо- вания	C++	C#	Python	Delphi	Visual Basic	Веса
C++	1	3	3	1/3	1/4	0.15
C#	1/3	1	1/3	1/4	1/4	0.06
Python	1/3	3	1	1/3	1/4	0.1
Delphi	3	4	3	1	2	0.37
Visual Basic	4	4	4	1/2	1	0.32
ИС = 9.7%			ОС = 8.7%			

систем, высокопроизводительных серверов, а также развлекательных приложений (например, видеоигры). Существует несколько реализаций языка C++ — как бесплатных, так и коммерческих. Их производят Проект GNU, Microsoft, Intel и Embarcadero (Borland). C++ оказал огромное влияние на другие языки программирования, в первую очередь на Java и C#.

При создании C++ стремились сохранить совместимость с языком C.

Множество программ, которые могут одинаково успешно транслироваться как компиляторами С, так и компиляторами С++, довольно велико — отчасти благодаря тому, что синтаксис С++ был основан на синтаксисе С.

Таблица 2.7 — Веса альтернатив относительно критерия «Скорость разработки»

Скорость разработки	С++	С#	Python	Delphi	Visual Basic	Веса
С++	1	2	2	1/5	1/4	0.12
С#	1/2	1	1/2	1/3	1/3	0.08
Python	1/2	2	1	1/4	1/3	0.1
Delphi	5	3	4	1	1/2	0.32
Visual Basic	4	3	3	2	1	0.38
ИС = 8.7%			ОС = 7.8%			

Таблица 2.12 — Веса альтернатив относительно критерия «Кроссплатформенность»

Кросс плат- формен- ность	С++	С#	Python	Delphi	Visual Basic	Веса
С++	1	1/4	1/2	1/6	1/6	0.05
С#	4	1	3	1/3	1/3	0.16
Python	2	1/3	1	1/4	1/4	0.09
Delphi	6	3	4	1	1	0.35
Visual Basic	6	3	4	1	1	0.35
ИС = 2.4%			ОС = 2.1%			

Стандарт С++ на 2003 год состоит из двух основных частей: описание ядра языка и описание стандартной библиотеки.

Кроме того, существует огромное количество библиотек C++, не входящих в стандарт. В программах на C++ можно использовать многие библиотеки C.

Стандартизация определила язык программирования C++, однако за этим названием могут скрываться также неполные, ограниченные, нестандартные варианты языка. Первое время язык развивался вне формальных рамок, спонтанно, по мере встававших перед ним задач. Развитию языка сопутствовало развитие кросс-компилятора cfront. Новшества в языке отражались в изменении номера версии кросс-компилятора. Эти номера версий кросс-компилятора распространялись и на сам язык, но применительно к настоящему времени речь о версиях языка C++ не ведут.

C++ добавляет к C объектно-ориентированные возможности. Он вводит классы, которые обеспечивают три самых важных свойства ООП: инкапсуляцию, наследование и полиморфизм.

Текущий стандарт языка ISO/IEC 14882:2003(E) был принят в 2003 году. Неофициально его обозначают как C++03. Следующая версия стандарта имеет неофициальное обозначение C++0x.

C++ продолжает развиваться, чтобы отвечать современным требованиям. Одна из групп, занимающихся языком C++ в его современном виде и направляющих комитету по стандартизации C++ советы по его улучшению — это Boost. Например, одно из направлений деятельности этой группы — совершенствование возможностей языка путём добавления в него особенностей метапрограммирования.

Стандарт C++ не описывает способы именования объектов, некоторые детали обработки исключений и другие возможности, связанные с деталями реализации, что делает несовместимым объектный код, созданный различными компиляторами. Однако для этого третьими лицами создано множество стандартов для конкретных архитектур и операционных систем.

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

Тем не менее среди компиляторов C++ всё ещё продолжается битва за полную реализацию стандарта C++, особенно в области шаблонов — части языка, совсем недавно полностью разработанной комитетом стандартизации.

Для поддержки кроссплатформенности будет использоваться библиотека Qt.

Qt — кросс-платформенный инструментальный разработки ПО на языке программирования C++. Есть также «привязки» ко многим другим языкам программирования: Python — PyQt, PySide; Ruby — QtRuby[6] ; Java — Qt Jambi[7]; PHP — PHP-Qt и другие.

Позволяет запускать написанное с его помощью ПО в большинстве современных операционных систем путём простой компиляции программы для каждой ОС без изменения исходного кода. Включает в себя все основные классы, которые могут потребоваться при разработке прикладного программного обеспечения, начиная от элементов графического интерфейса и заканчивая классами для работы с сетью, базами данных и XML. Qt является полностью объектно-ориентированным, легко расширяемым и поддерживающим технику компонентного программирования.

Отличительная особенность Qt от других библиотек — использование Meta Object Compiler (MOC) — предварительной системы обработки исходного кода (в общем-то, Qt — это библиотека не для чистого C++, а для его особого наречия, с которого и «переводит» MOC для последующей компиляции любым стандартным C++ компилятором). MOC позволяет во много раз увеличить мощь библиотек, вводя такие понятия, как слоты и сигналы. Кроме того, это позволяет сделать код более лаконичным.

Для работы с видео-камерой воспользуемся библиотекой OpenCV.

OpenCV (англ. Open Source Computer Vision Library, библиотека компьютерного зрения с открытым исходным кодом) — библиотека алгоритмов

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		

компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом. Реализована на C/C++, также разрабатывается для Python, Ruby, Matlab, Lua и других языков[8]. Может свободно использоваться в академических и коммерческих целях — распространяется в условиях лицензии BSD.

Для увеличения скорость работы будут использованы параллельные вычисления.

OpenMP (Open Multi-Processing) — открытый стандарт для распараллеливания программ на языках Си, Си++ и Фортран. Описывает совокупность директив компилятора, библиотечных процедур и переменных окружения, которые предназначены для программирования многопоточных приложений на многопроцессорных системах с общей памятью.

OpenMP реализует параллельные вычисления с помощью многопоточности, в которой «главный» (master) поток создает набор подчиненных (slave) потоков и задача распределяется между ними. Предполагается, что потоки выполняются параллельно на машине с несколькими процессорами (количество процессоров не обязательно должно быть больше или равно количеству потоков).

					Выбор и обоснование средств проектирования	Лист
Изм.	Лист	№ докум.	Подпись	Дата		