

ISSN 2310-3353



«А. БАЙТҰРСЫНОВ  
АТЫНДАҒЫ ҚОСТАНАЙ ӨңІРЛІК  
УНИВЕРСИТЕТІ» КЕАҚ



# ҚМПИ ЖАРШЫСЫ

ҒЫЛЫМИ-ӘДІСТЕМЕЛІК ЖУРНАЛ  
НАУЧНО-МЕТОДИЧЕСКИЙ ЖУРНАЛ

№ 2  
2023



**PUBLISHINGS**  
**K S P I**



**Қ М П И**  
**ЖАРШЫСЫ**

**ВЕСТНИК**  
**К Г П И**

2023 ж., сәуір, №2 (70)  
Журнал 2005 ж. қаңтардан бастап шығады  
Жылына төрт рет шығады

**Құрылтайшы: А. Байтұрсынов атындағы Қостанай өңірлік университеті**

**Бас редактор: Қуанышбаев С. Б.**, география ғылымдарының докторы, А. Байтұрсынов атын. ҚӨУ, Қазақстан

**Бас редактордың орынбасары: Жарлығасов Ж.Б.**, ауыл шаруашылығы ғылымдарының кандидаты, А. Байтұрсынов атын. ҚӨУ, Қазақстан

#### **РЕЦЕНЗЕНТТЕР**

**Березнова Е.В.**, педагогика ғылымдарының докторы, ММХҚИ СИМ, Мәскеу қ., Ресей

**Жаксылыкова К.Б.**, педагогика ғылымдарының докторы, Қ. Сәтпаев атындағы Қазақ ұлттық техникалық зерттеу университетінің профессоры, Қазақстан

#### **РЕДАКЦИЯ АЛҚАСЫ**

**Амирова Б.А.**, психология ғылымдарының докторы, Е.А. Букетов атын. ҚарМУ, Қазақстан

**Благодарзумная О.Н.**, экономика ғылымдарының кандидаты, Молдова Халықаралық Тәуелсіз Университетінің доценті, Молдова

**Доман Э.**, лингвистикалық ғылымдар докторы, Макао университеті, Сидней, Австралия

**Елагина В.С.**, педагогика ғылымдарының докторы, профессор, ООМГПУ, Ресей

**Жилбаев Ж.О.**, педагогика ғылымдарының кандидаты, доцент, Ы. Алтынсарин атындағы Ұлттық білім академиясы президенті, Қазақстан

**Кайе Ж.**, философия ғылымдарының докторы, Виа Домисия Университетінің профессоры, Перпиньян қ., Франция

**Катцнер Т.**, Батыс Вирджиния Университетінің профессоры, PhD докторы, АҚШ, Батыс Вирджиния

**Кульгильдинова Т.А.**, педагогика ғылымдарының докторы, Абылай хан атындағы ҚазХҚ және ӨТУ-нің профессоры, Қазақстан

**Марилена Сантана дос Сантос Гарсия**, лингвистикалық ғылымдар докторы, Сан-Паулу Папа католик университеті, Бразилия

**Монова-Желева М.**, PhD докторы, Бургас еркін университетінің профессоры, Болгария

**Чаба Толгизи**, Венгрияның Сегед Университеті экология кафедрасының ғылыми қызметкері, Венгрия

Тіркеу туралы куәлік №5452-Ж

Қазақстан Республикасының ақпарат министрлігімен 17.09.2004 берілген.

Мерзімді баспа басылымын қайта есепке алу 29.03.2021 ж.

Жазылу бойынша индексі 74081

#### **Редакцияның мекен-жайы:**

110000, Қостанай қ., Тәуелсіздік к., 118

(Ғылым және коммерциализация басқармасы)

Тел. (7142) 54-58-74 (160)

© А. Байтұрсынов атындағы Қостанай өңірлік университеті

7. Donnelly, D., McGarr, O., & O'Reilly, J. (2011). A Framework for teachers' integration of ICT into their classroom practice. *Computers & Education*, 57(2), 1469–1483. <https://doi.org/10.1016/j.compedu.2011.02.014>

8. Drijvers, P. (2013). Digital technology integration in mathematics education: Why it works (or doesn't). *PNA*, 8(1), 1–20. [https://doi.org/10.1007/978-3-319-17187-6\\_8](https://doi.org/10.1007/978-3-319-17187-6_8).

**АМАНБАЙ, А.А., ЗАУРБЕКОВ, Н.С.**

#### **GIMKIT: ГЕЙМИФИКАЦИЯ МАТЕМАТИКИ НА ОСНОВЕ ВИКТОРИНЫ**

*Математика очень важна в нашей жизни и обществе. Однако геймификация математики встречается редко. Это исследование предлагает разработку обучающей математической игры под названием Gimkit. Это новый инструмент формативной оценки, основанный на игре, который использует несколько стратегий, чтобы «заработать» как можно больше денег, отвечая на вопросы викторины. Это исследование направлено на изучение геймификации. Тестирование проводилось среди учащихся средних школ (n=24; возраст=12,13) села Айтей Карасайского района Алматинской области, Республики Казахстан. Учащимся было предложено ответить на предварительные и послетестовые вопросы. Результаты многообещающие, где анализ показал значительную разницу между результатами до тестирования (M=13,7, SD=2,54) и после тестирования (M=16,5, SD=2,75). Таким образом, геймификация математики с помощью игры-викторины Gimkit повышает их математические знания и память.*

**Ключевые слова:** игровое обучение, интерактивное обучение, математические знания, мониторинг успеваемости учащихся.

**AMANBAY, A.A., ZAURBEKOV, N.S.**

#### **GIMKIT: QUIZ-BASED GAMIFICATION OF MATH**

*Mathematics is very important in our life and society. However, gamification of mathematics is rare. This research suggests the development of an educational math game called Gimkit. This is a new formative assessment tool based on a game that uses several strategies to "earn" as much money as possible by answering quiz questions. This study is aimed at studying gamification. The testing was conducted among secondary school students (n=24; age=12,13) of the village of Aitei, Karasai district, Almaty region, Republic of Kazakhstan. Students were asked to answer preliminary and post-text questions. The results are promising, where the analysis showed a significant difference between the results before testing (M=13.7, SD=2.54) and after testing (M=16.5, SD=2.75). Thus, gamification of mathematics with the help of the Gimkit quiz game increases their mathematical knowledge and memory.*

**Key words:** game learning, interactive learning, mathematical knowledge, monitoring of student progress.

**УДК 004.432.45**

**Макамбетова, К.М.,**

магистрант 2 курса КРУ им. А. Байтурсынова  
г.Костанай, Казахстан

### **ВЫБОР ЯЗЫКА ПРОГРАММИРОВАНИЯ ДЛЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОМПЬЮТЕРНОЙ СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ**

#### **Аннотация**

*В течение нескольких последних десятилетий сфера информационных и компьютерных технологий шагнула далеко вперед. Технологии прочно вошли в нашу повседневную жизнь, и уже довольно сложно представить современного человека без обработки какой-либо информации на компьютере. Программирование, само собой, идет бок о бок с техническим прогрессом и развивается столь же стремительно, как и остальные технологии. Развиваются старыязыки программирования,*

*разрабатываются новые. За всю историю развития электронно-вычислительных машин появились сотни языков программирования, многие из которых используются и по сей день. В статье рассматриваются языки программирования высокого уровня для разработки программного обеспечения компьютерной системы автоматизированного тестирования.*

***Ключевые слова:** языки программирования, высокоуровневые языки программирования, информационные технологии, классификации языков программирования, критерии выбора языка программирования.*

## **1 Введение**

Прогресс развития компьютерных технологий определил процесс появления новых разнообразных знаковых систем для записи алгоритмов – языков программирования. Язык программирования – язык (знаковая система), предназначенный для записи компьютерных программ; определяет набор лексических, синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно – ЭВМ) под ее управлением.

Со времени создания первых программируемых машин человечество придумало более 8 тыс. языков программирования (включая нестандартные, визуальные и эзотерические языки) [1]. Каждый год их число увеличивается. Некоторыми языками умеет пользоваться только небольшое число их собственных разработчиков, другие становятся известны миллионам людей.

Язык программирования предназначен для написания компьютерных программ, которые представляют собой набор правил, позволяющих компьютеру выполнить тот или иной вычислительный процесс, организовать управление различными объектами, и т.п. [2]. Язык программирования отличается от естественных языков тем, что предназначен для взаимодействия человека с ЭВМ, в то время как естественные языки используются для общения людей между собой.

Большинство языков программирования использует специальные конструкции для определения и манипулирования структурами данных и управления процессом вычислений.

На данный момент существует большое количество языков программирования – компилируемые и интерпретируемые языки. Каждый язык создавался с определенной целью. Выбор языка программирования для разработки программного обеспечения является очень важным этапом, от которого зависит скорость создания программы, скорость тестирования, возможность переноса на другие платформы, возможность быстрого внесения изменений, быстрота выполнения конечного продукта и т.д. При этом стоит помнить, что идеального языка не существует, все они обладают своими положительными и отрицательными качествами, которые будут так или иначе влиять на процесс разработки.

## **2 Материалы и методы**

С целью выбора языка программирования для разработки программного обеспечения компьютерной системы автоматизированного тестирования мы применяли теоретические методы и эмпирические методы. Так, метод теоретического анализа научно-педагогической и методической литературы позволил выявить критерии выбора языка программирования для разработки программного обеспечения компьютерной системы автоматизированного тестирования, а также на основе выявления особенностей традиционных языков программирования высокого уровня. Использован метод сравнения для сопоставления возможностей языков программирования с критериями, выявленными нами для разработки программного обеспечения компьютерной системы автоматизированного тестирования.

## **3,4 Результаты и обсуждение**

В современной классификации языков программирования по уровням всё чаще используют деление на два уровня: языки высокого и низкого уровней [2]. Языками низкого уровня считаются языки, близкие к машинному коду и ориентированные на конкретные

команды процессора, то есть машинно-ориентированные языки, а языками высокого уровня в современной классификации считаются машинно-независимые языки, разработанные для удобства использования.

Используя названные выше методы мы выявили критерии для выбора языка программирования для разработки программного обеспечения компьютерной системы автоматизированного тестирования.

1. Скорость работы конечного продукта.

Требовательным к скорости выполнения могут быть программы с большим объемом математических вычислений, например моделирование физических систем, расчеты большого объема экономических данных, выведение трехмерной графики и прочее. Для данных целей хорошо подойдут компилируемые языки: *PHP*, *C/C++*, *фортран* и т.д.

2. Объем занимаемой оперативной памяти.

Данное требование появляется, когда программа разрабатывается для встраиваемых систем, мобильных платформ, микроконтроллеров и так далее. В данных случаях, чем меньше памяти расходует программа на данном языке – тем лучше. К таким языкам, опять же, относятся *PHP*, *C/C++*, *Objective-C* и другие. Список языков подобен списку пункта 1, так как чем меньше функциональных блоков в схеме исполнения, тем меньше занимает и памяти компьютера.

Если данное требование не критично, то можно использовать «истинно высокоуровневые языки».

3. Скорость разработки программы.

Данное требование возникает тогда, когда нужно быстро и оперативно разработать программное обеспечение. Тогда выбор падает на высокоуровневые языки с максимально простым синтаксисом. Это, например, *Java*, *Flash*, *PHP* и подобные. На данных языках время разработки может существенно сокращаться из-за обилия сторонних библиотек, максимально простого синтаксиса. Скорость выполнения программ, написанных на данных языках страдает, причем порой весьма ощутимо. Схема выполнения на примере *Java*:

*Программа в виде байт-кода -> виртуальная машина-анализатор -> системные библиотеки -> драйвера -> железо.*

Самым медленно работающим блоком в данной схеме является анализатор – он должен байт-код программы транслировать «на лету» в машинный код, при этом потратив много времени на точное определение инструкции. Поэтому быстрая разработка зачастую означает медленное выполнение.

4. Ориентированность на компьютер или человека.

С кем будет работать программа в первую очередь? С человеком, или с компьютером? В первом случае программа должна обладать мощной графической частью, отвечающей требованиям дизайна и понятным интерфейсом. Разработка графической части зачастую требует достаточно много времени, т.к. отличается немалой сложностью. Здесь сложность возникает в том, что вывод графики – это немало математики, а значит присутствует требовательность к скорости исполнения, а из-за сложности разработки присутствует необходимость в высокоуровневом языке. В данном случае, на наш взгляд, очень хорошо подходит *PHP*, *C++/C#* с их одновременной и высокоуровневостью, и скоростью выполнения программ на них. Если программа ориентирована в первую очередь на «скрытую работу» с минимумом взаимодействия с пользователем, тогда выбор должен ложиться в сторону быстрых языков (*ASM*, *C*)

5. Кроссплатформенность.

Кроссплатформенность – возможность работы программы на различных платформах, в различных ОС с минимальными изменениями. В этой сфере можно выделить такие языки: *Java*, *PHP*, *C#,Flash,C++* с различными библиотеками и другие, менее используемые, языки. *Java* создавался с тем условием, что программы на данном языке должны работать на любой платформе, где есть *JVM – Java Virtual Machine*. Программы на *Java* вообще не требуют



никаких изменений – после компиляции получается .jar файл, который будет работать и на Windows, и на Mac OS, и на Linux и еще немало где. Аналогичная ситуация и с Flash, только список платформ гораздо менее обширный. С C++ дело обстоит труднее. На чистом C++ написать кроссплатформенную программу довольно трудно, у кода возникает обширная избыточность, теряется достоинство в скорости выполнения. Облегчают задачу кроссплатформенные библиотеки, например, Qt, которые позволяют добиться принципа «один код на все платформы», однако на каждую платформу нужно программу собирать отдельно (при этом разными компиляторами).

В этот раздел так же можно включить интерпретируемые, скриптовые языки – для их работы нужно наличие интерпретатора языка в системе. Данные языки очень удобны в плане разработки, но достаточно медлительны. Схема их работы напоминает схему работы Java/Flash, только анализатор стал еще медленнее – полумашинный байт код анализировать «на лету» гораздо проще, чем человеческий код. Также, это влечет к большему потреблению памяти.

#### 6. Скорость внесения изменений, скорость тестирования.

Проект стремительно развивается, в него постоянно вносятся изменения, порой немало? Тогда выбор должен падать на высокоуровневые языки, где любой функциональный блок можно быстро переписать. Для подтверждения гораздо проще использовать C++, чем ассемблер. А еще проще Java. Но тут очень много тонкостей, которые таятся даже не сколько в языке, сколько в разработчике с его стилем программирования и компиляторах. Тем не менее, язык вносит свою долю в это дело, так или иначе упрощая/осложняя работу программиста.

Таким образом, проанализировав критерии выбора языка программирования, мы выделили критерии, которые подходят для разработки программного обеспечения компьютерной системы автоматизированного тестирования:

1. Скорость работы конечного продукта. Данный критерий был выбран, так как после сборки программа не требует ничего лишнего и содержит в себе машинные команды, которые выполняются без лишних задержек. Схема работы таких программ такая:

1) программа исполняется сразу, так сказать она самодостаточна и не требует дополнительных библиотек;

2) программа кроме своего кода содержит вызовы библиотек с машинным кодом (как системных, так и входящих в проект), поэтому, кроме исполнения собственно своих команд, программа вызывает функции из библиотек;

3) в дополнение случаям 1 и 2, программа может работать через прослойку драйверов, которые написаны на языках низкого уровня и работают по умолчанию быстро. Как видно, максимум в схеме возможны 4 блока: *программа -> библиотеки -> драйвера -> железо*.

2. Ориентированность на человека. Так как программное обеспечение компьютерной системы автоматизированного тестирования будет использоваться учащимися школы и учителями был выбран данный критерий. Программное обеспечение должно обладать хорошей графической частью и понятным пользовательским интерфейсом.

#### 3. Скорость внесения изменений, скорость тестирования.

Таким образом, под выделенные нами критерии подходят такие языки программирования, как C, C++, Java, PHP. Рассмотрим каждый язык программирования отдельно:

C – компилируемый статически типизированный язык программирования общего назначения, разработанный в 1969–1973 годах сотрудником Bell Labs Деннисом Ритчи как развитие языка Би. Первоначально был разработан для реализации операционной системы UNIX, но впоследствии был перенесён на множество других платформ [3]. Согласно дизайну языка, его конструкции близко сопоставляются типичным машинным инструкциям, благодаря чему он нашёл применение в проектах, для которых был свойственен язык ассемблера, в том числе как в операционных системах, так и в различном прикладном

программном обеспечении для множества устройств – от суперкомпьютеров до встраиваемых систем.

Основные особенности С:

- простая языковая база, из которой в стандартную библиотеку вынесены многие существенные возможности, вроде математических функций или функций работы с файлами;

- ориентация на процедурное программирование;
- система типов, предохраняющая от бессмысленных операций;
- использование препроцессора для абстрагирования однотипных операций;
- доступ к памяти через использование указателей;
- небольшое число ключевых слов;
- передача параметров в функцию по значению, а не по ссылке (передача по ссылке эмулируется с помощью указателей);
- наличие указателей на функции и статические переменные;
- области видимости имён;
- структуры и объединения – определяемые пользователем собирательные типы данных, которыми можно манипулировать как одним целым.

В то же время в С отсутствуют:

- вложенные функции;
- прямое возвращение нескольких значений из функций;
- сопрограммы;
- средства автоматического управления памятью;
- встроенные средства объектно-ориентированного программирования;
- средства функционального программирования.

С++ – компилируемый, статически типизированный язык программирования общего назначения.

Поддерживает такие парадигмы программирования, как процедурное программирование, объектно-ориентированное программирование, обобщённое программирование. Язык имеет богатую стандартную библиотеку, которая включает в себя распространённые контейнеры и алгоритмы, ввод-вывод, регулярные выражения, поддержку многопоточности и другие возможности. С++ сочетает свойства как высокоуровневых, так и низкоуровневых языков [4]. В сравнении с его предшественником – языком С – наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования

С++ широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также компьютерных игр.

Синтаксис С++ унаследован от языка С. Изначально одним из принципов разработки было сохранение совместимости с С. Тем не менее С++ не является в строгом смысле надмножеством С; множество программ, которые могут одинаково успешно транслироваться как компиляторами С, так и компиляторами С++, довольно велико, но не включает все возможные программы на С.

Java – строго типизированный объектно-ориентированный язык программирования общего назначения, разработанный компанией Sun Microsystems (в последующем приобретённой компанией Oracle). Разработка ведётся сообществом, организованным через Java Community Process; язык и основные реализующие его технологии распространяются по лицензии GPL. Права на торговую марку принадлежат корпорации Oracle [4].

Приложения Java обычно транслируются в специальный байт-код, поэтому они могут работать на любой компьютерной архитектуре, для которой существует реализация виртуальной Java-машины. Дата официального выпуска — 23 мая 1995 года. Занимает

высокие места в рейтингах популярности языков программирования (2-е место в рейтингах IEEE Spectrum (2020)[3] и TIOBE (2021) [4]).

Программы на Java транслируются в байт-код Java, выполняемый виртуальной машиной Java (JVM) – программой, обрабатывающей байтовый код и передающей инструкции оборудованию как интерпретатор.

Достоинством подобного способа выполнения программ является полная независимость байт-кода от операционной системы и оборудования, что позволяет выполнять Java-приложения на любом устройстве, для которого существует соответствующая виртуальная машина. Другой важной особенностью технологии Java является гибкая система безопасности, в рамках которой исполнение программы полностью контролируется виртуальной машиной. Любые операции, которые превышают установленные полномочия программы (например, попытка несанкционированного доступа к данным или соединения с другим компьютером), вызывают немедленное прерывание.

Часто к недостаткам концепции виртуальной машины относят снижение производительности. Ряд усовершенствований несколько увеличил скорость выполнения программ на Java:

- применение технологии трансляции байт-кода в машинный код непосредственно во время работы программы (JIT-технология) с возможностью сохранения версий класса в машинном коде,
- обширное использование платформенно-ориентированного кода (native-код) в стандартных библиотеках,
- аппаратные средства, обеспечивающие ускоренную обработку байт-кода (например, технология Jazelle, поддерживаемая некоторыми процессорами архитектуры ARM).

Язык программирования PHP.

Термин PHP является аббревиатурой от *PHP: препроцессор гипертекста*. PHP – это серверный язык сценариев, разработанный специально для веб-разработки [5]. Он с открытым исходным кодом, что означает, что его можно бесплатно загрузить и использовать. Его очень просто изучить и использовать. Файлы имеют расширение «.php».

Расмус Лерддорф вдохновил первую версию PHP и участвовал в более поздних версиях. Это интерпретируемый язык, и он не требует компилятора.

- Код PHP выполняется на сервере.
- Его можно интегрировать со многими базами данных, такими как Oracle, Microsoft SQL Server, MySQL, PostgreSQL, Sybase и Informix.
- Это мощная система управления контентом, такая как WordPress, и ее можно использовать для контроля доступа пользователей.
- Он поддерживает основные протоколы, такие как HTTP Basic, HTTP Digest, IMAP, FTP и другие.
- Такие веб-сайты, как [www.facebook.com](http://www.facebook.com) и [www.yahoo.com](http://www.yahoo.com), также построены на PHP. Одна из основных причин этого заключается в том, что PHP можно легко встроить в файлы HTML, а коды HTML также можно записать в файл PHP.
- Вещь, которая отличает PHP от клиентского языка, такого как HTML, заключается в том, что коды PHP выполняются на сервере, тогда как коды HTML отображаются непосредственно в браузере. Коды PHP сначала выполняются на сервере, а затем результат возвращается в браузер.
- Единственная информация, которую знает клиент или браузер, – это результат, возвращаемый после выполнения скрипта PHP на сервере, а не фактические коды PHP, присутствующие в файле PHP. Кроме того, файлы PHP могут поддерживать другие языки сценариев на стороне клиента, такие как CSS и JavaScript.

Название PHP расшифровывается как Hypertext Preprocessor и обозначает язык сценариев на стороне сервера, который предполагает, что приложения, написанные на нем, работают на веб-серверах и не зависят от онлайн-браузера. Синтаксис языка PHP аналогичен



языку Си. Он создан Расмусом Лердорфом и появился в 1995 году. PHP широко используется при разработке веб-приложений и стал одним из основных языков для разработчиков, создающих новые приложения.

Ведущие сайты социальных сетей, такие как Facebook, и известные организации, такие как Гарвардский университет, используют PHP, что делает PHP популярным и повышает доверие к нему. Однако с годами область его использования изменилась, и в настоящее время язык программирования PHP считается одним из самых простых и популярных инструментов программирования для веб-разработки благодаря своим многочисленным достоинствам, которые могут быть главной целью этого текста. Это считается действительно эффективной технологией, которая обеспечивает удобный процесс разработки с множеством дополнительных инструментов, помогающих ему. Фактически, согласно индексу языков программирования (PYPL), PHP является пятым по популярности языком программирования в мире.

PHP широко используется при разработке веб-приложений и других приложений во всех областях. Несколько технологий, разработка которых поддерживается PHP, перечислены ниже:

- Система управления контентом.
- Веб-приложения и разработка сайтов.
- Интернет-магазины и приложения.
- Аналитика данных и представление.
- Обработка изображений.
- Приложения, основанные на графическом интерфейсе.
- Развитие возможностей Flash.

Преимущества PHP:

- Наиболее важным преимуществом PHP является то, что он является бесплатным и открытым исходным кодом. Его можно загрузить где угодно и легко использовать для мероприятий или веб-приложений.

- Он не зависит от платформы. Приложения на основе PHP могут работать в любой ОС, такой как UNIX, Linux, Windows и т. д.

- Можно легко загружать приложения, основанные на PHP и подключенные к базе данных. Он в основном используется из-за более высокой скорости загрузки при низкой скорости интернета, чем другие языки программирования.

- У него меньше кривых обучения, потому что он прост и понятен в использовании.

Кто-то, кто знаком с программированием на C, может легко работать с PHP.

- Он более стабилен в течение нескольких лет благодаря постоянной поддержке различных версий.

- Это помогает повторно использовать эквивалентный код и избавляет от необходимости писать длинный код и сложную структуру для событий веб-приложений.

- Это помогает легко управлять кодом.

- Он имеет мощную библиотечную поддержку для использования различных функциональных модулей для представления данных.

- Встроенные в PHP модули подключения к базам данных помогают легко подключать базы данных, сокращая количество проблем и время, необходимое для разработки веб-приложений и сайтов с контентом.

- По-пуляриность PHP привела к возникновению различных сообществ разработчиков, часть которых может быть потенциальными кандидатами на работу.

- Гибкость делает PHP готовым к эффективному сочетанию со многими другими языками программирования, чтобы программный пакет мог использовать самые эффективные технологии для каждой конкретной функции.

## **5 Выводы**

Таким образом, выделив основные критерии для разработки программного обеспечения компьютерной системы автоматизированного тестирования, и проанализировав преимущества и недостатки каждого языка программирования, мы пришли к выводу, что под наши критерии подходит язык программирования PHP.

## **Список литературы**

1. Макконнелл С. Совершенный код. – Москва: Русская редакция, 2017. – с. 56-59.
2. Языки программирования и их классификация [Электронный ресурс]. URL: <http://www.maksakov-sa.ru/ProgrProd/YazProgr/index.html>
3. Язык программирования. Эволюция языков программирования [Электронный ресурс]. URL: [http://wiki.mvтом.ru/index.php/Язык\\_программирования](http://wiki.mvтом.ru/index.php/Язык_программирования).
4. Эволюция языков программирования [Электронный ресурс]. URL: [http://www.urtt.ru/bib/dataindex/oaip/lection/html/lect\\_05.html](http://www.urtt.ru/bib/dataindex/oaip/lection/html/lect_05.html)
5. Пять поколений языков программирования [Электронный ресурс]. URL: [http://life-prog.ru/view\\_zam2.php?id=194&cat=5&page=11](http://life-prog.ru/view_zam2.php?id=194&cat=5&page=11)

## **МАКАМБЕТОВА, К.М.**

### **КОМПЬЮТЕРЛІК ТЕСТІЛЕУ ЖҮЙЕСІНЕ БАҒДАРЛАМАЛЫҚ ҚАМТАМАСЫЗ ЕТУДІ ӨЗІРЛЕУ ҮШІН БАҒДАРЛАМАЛАУ ТІЛІН ТАҢДАУ**

*Соңғы бірнеше онжылдықта ақпараттық-компьютерлік технологиялар саласы біршама алға қадам басты. Технология біздің күнделікті өмірімізге мықтап енді және қазіргі заманғы адамды компьютерде ешқандай ақпаратты өңдемей елестету қазірдің өзінде өте қиын. Бағдарламалау, әрине, техникалық прогреспен қатар жүреді және басқа технологиялар сияқты тез дамиды. Ескі бағдарламалау тілдері дамып келеді, жаңалары жасалуда. Электрондық есептеуіш машиналардың даму тарихында жүздеген бағдарламалау тілдері пайда болды, олардың көпшілігі бүгінгі күнге дейін қолданылады. Мақалада компьютерлік тестілеу жүйесіне арналған бағдарламалық қамтамасыз етуді әзірлеуге арналған жоғары деңгейлі бағдарламалау тілдері қарастырылады.*

**Кілт сөздер:** бағдарламалау тілдері, жоғары деңгейлі бағдарламалау тілдері, ақпараттық технологиялар, бағдарламалау тілдерінің классификациясы, бағдарламалау тілін таңдау критерийлері.

## **МАКАМБЕТОВА, К.М.**

### **CHOOSING A PROGRAMMING LANGUAGE FOR DEVELOPING SOFTWARE FOR A COMPUTER-AIDED TESTING SYSTEM**

*Over the past few decades, the field of information and computer technology has stepped far forward. Technology has firmly entered our daily life, and it is already quite difficult to imagine a modern person without processing any information on a computer. Programming, of course, goes hand in hand with technical progress and develops as rapidly as other technologies. Old programming languages are developing, new ones are being developed. Throughout the history of the development of electronic computers, hundreds of programming languages have appeared, many of which are still used today. The article deals with high-level programming languages for developing software for a computer-aided testing system.*

**Key words:** programming languages, high-level programming languages, information technology, classification of the programming languages, criteria for choosing any programming language.