



<http://dx.doi.org/10.35596/1729-7648-2023-29-4-41-49>

Оригинальная статья
Original paper

УДК 004.042

АЛГОРИТМ РАЗРАБОТКИ И МЕТОДИКА ИСПОЛЬЗОВАНИЯ В УЧЕБНОМ ПРОЦЕССЕ ПРОГРАММНОГО СРЕДСТВА ДЛЯ ПРЕОБРАЗОВАНИЯ ИНФОРМАЦИИ

В. А. ГЕРАСИМОВ¹, О. В. БОЙПРАВ²

¹Научно-исследовательский институт технической защиты информации
(г. Минск, Республика Беларусь)

²Белорусский государственный университет информатики и радиоэлектроники
(г. Минск, Республика Беларусь)

Поступила в редакцию 16.02.2023

© Белорусский государственный университет информатики и радиоэлектроники, 2023
Belarusian State University of Informatics and Radioelectronics, 2023

Аннотация. Статья посвящена результатам обоснования и создания алгоритма разработки программного средства для преобразования информации, которое целесообразно задействовать в процессе цифровой трансформации образования в высших учебных заведениях. В ходе обоснования этого алгоритма определена роль изучения вопросов, касающихся преобразования информации, в формировании профессиональных компетенций будущих специалистов в сфере информационных технологий; определены преимущества использования языка C по сравнению с другими языками программирования для разработки программного средства; рассмотрены возможности работы с памятью при помощи указателей на языке программирования C. Представлены ключевые фрагменты кода программного средства, разработанного в соответствии с предложенным алгоритмом, связанные с преобразованием текстовой информации в числовые данные и числовых данных в текстовую информацию. Продемонстрированы примеры работы этого программного средства и представлены пути его использования при проведении лабораторных или практических работ по дисциплинам, тематика которых связана с информационными технологиями, со студентами 1-го и 2-го курсов. Определены пути усовершенствования предложенного алгоритма, направленные на увеличение количества учебных дисциплин, в рамках которых целесообразно использовать разработанное программное средство.

Ключевые слова: языки программирования, образование, указатели, работа с памятью, позиционные системы счисления, межпредметные связи, преобразование информации.

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Для цитирования. Герасимов, В. А. Алгоритм разработки и методика использования в учебном процессе программного средства для преобразования информации / В. А. Герасимов, О. В. Бойправ // Цифровая трансформация. 2023. Т. 29, № 4. С. 41–49. <http://dx.doi.org/10.35596/1729-7648-2023-29-4-41-49>.

ALGORITHM OF DEVELOPMENT AND METHODOLOGY OF USING SOFTWARE FOR INFORMATION CONVERSION IN THE EDUCATIONAL PROCESS

VYACHESLAV A. GERASIMOV¹, OLGA V. BOYPRAV²

¹Scientific Research Institute of Technical Protection of Information (Minsk, Republic of Belarus)

²Belarusian State University of Informatics and Radioelectronics (Minsk, Republic of Belarus)

Submitted 16.02.2023

Abstract. The article is devoted to the results of substantiation and creation of the algorithm for the development of a software tool for information transformation, which is advisable to be used in the process of digital transformation of education in higher educational institutions. In the course of substantiating this algorithm, the role of studying issues related to the information transformation in the formation of professional competencies of future specialists in the field of information technology is determined; the advantages of using the C language in comparison with other programming languages for software development are determined; the possibilities of working with memory using pointers in the C programming language are considered. The key code fragments of a software tool developed in accordance with the proposed algorithm related to the transformation of textual information into numeric data and numeric data into textual information are presented. Examples of the operation of this software tool are demonstrated and the ways for its use is presented when conducting laboratory or practical work in disciplines whose topics are related to information technology, with students of the 1st and 2nd years of study. The ways of improving the proposed algorithm aimed at increasing the number of academic disciplines, within which it is advisable to use the software developed in accordance with it, are determined.

Keywords: programming languages, education, pointers, working with memory, positional number systems, interdisciplinary communications, information transformation.

Conflict of interests. The authors declare no conflict of interests.

For citation. Gerasimov V. A., Boyprav O. V. (2023) Algorithm of Development and Methodology of Using Software for Information Conversion in the Educational Process. *Digital Transformation*. 29 (4), 41–49. <http://dx.doi.org/10.35596/1729-7648-2023-29-4-41-49> (in Russian).

Введение

В настоящее время наблюдается тенденция увеличения количества информации, предоставляемой в электронной форме, что обусловлено цифровой трансформацией промышленности, сельского хозяйства, здравоохранения, образования. Эта трансформация связана с такими процессами, как внедрение системы электронного документооборота [1] в деятельность промышленных и сельскохозяйственных предприятий¹, разработка электронных карточек пациентов [2] и электронных рецептов в сфере здравоохранения², применение дистанционных образовательных технологий³ учителями школ и преподавателями средних специальных и высших учебных заведений [3]. Динамика изменения удельного веса учреждений общего среднего образования, участвующих в проекте «Электронная школа», за 2016–2021 годы следующая⁴: 2016-й – 2,5 %, 2017 – 16,4 %, 2018 – 16,4 %, 2019 – 59 %, 2020 – 80 %, 2021-й – 85 %. По данным Национального статистического комитета Республики Беларусь⁴, прирост возможностей использования электронных карточек пациентов и электронных рецептов в сфере здравоохранения с 2016 по 2021 год составляет: 2016-й – 25 %, 2017 – 69,1 %, 2018 – 76 %, 2019 – 95,3 %, 2020 – 97,7 %, 2021-й – 97,2 %.

¹ Внедрение электронного документооборота в Беларуси: проблемы и пути их решения [Электронный ресурс]. Режим доступа: <https://elib.belstu.by/handle/123456789/44666>. Дата доступа: 26.01.2023.

² Электронное здравоохранение в контексте национальной безопасности Республики Беларусь [Электронный ресурс]. Режим доступа: https://libeloc.bsuir.by/bitstream/123456789/42753/1/Belyatskaya_Elektronnoye.pdf. Дата доступа: 26.01.2023.

³ Дистанционное образование в Беларуси: проблемы и перспективы [Электронный ресурс]. Режим доступа: <https://elib.bsu.by/bitstream/123456789/269815/1/529-534.pdf>. Дата доступа: 26.01.2023.

⁴ Национальные статистические показатели развития цифровой экономики в Республике Беларусь [Электронный ресурс]. Режим доступа: <https://www.belstat.gov.by/ofitsialnaya-statistika/makroekonomika-i-okruzhayushchaya-sreda/informatsionno-telekommunikatsionnye-tehnologii/tsifrovaya-ekonomika/>. Дата доступа: 26.01.2023.

Статистические данные за 2021 год в сфере межведомственного документооборота⁵ по областям в Республике Беларусь среди новых абонентов распределились следующим образом: Брестская область – 76,3 %, Витебская – 96,8 %, Гомельская – 99,6 %, Гродненская – 98,9 %, Минск – 149,9 %, Минская – 106,1 %, Могилевская область – 99,4 %.

В системе процессов, реализуемых в отношении информации, предоставляемой в электронной форме, важную роль играет процесс ее преобразования (кодирования)⁶. Это обусловлено тем, что данный процесс связан с оптимизацией памяти электронных носителей информации [4]. Поэтому знания о методах преобразования информации, хранимой на электронных носителях, являются ключевыми для формирования универсальных, базовых профессиональных и специализированных компетенций у будущих специалистов в сфере информационных технологий [5].

В связи с вышеизложенным в Белорусском государственном университете информатики и радиоэлектроники для студентов 1–2 курсов преподаются дисциплины, в рамках которых рассматриваются вопросы, связанные с преобразованием информации. К основным из таких дисциплин относятся: «Основы алгоритмизации и программирования», «Информационные системы и технологии», «Конструирование программного обеспечения», «Основы теории информации» [6]. Цель исследования, результаты которого представлены в статье, состояла в создании алгоритма разработки программного средства, которое целесообразно применять в ходе преподавания вышеперечисленных дисциплин для повышения наглядности теоретического материала, содержащего особенности преобразования информации (далее – программное средство).

Алгоритм разработки программного средства

Разработку программного средства целесообразно выполнять с помощью языка программирования C. Это обусловлено следующими его преимуществами по сравнению с другими популярными языками программирования⁷ (C#, Python, PHP, Java, JavaScript):

- использование при преподавании дисциплин для студентов 1–2 курсов университетов;
- невысокая продолжительность компиляции (интерпретации) программного кода;
- возможность работы с адресами памяти при помощи указателей без необходимости установки дополнительных компонентов или настройки интегрированной среды разработки.

Последнее из обозначенных преимуществ является наиболее существенным с точки зрения обоснования целесообразности использования языка C для разработки программного средства [7]. Данное преимущество обусловлено тем, что в языке программирования C с помощью указателей можно представлять символьную информацию (массив символов) в виде числовых данных путем явного приведения типов (рис. 1).

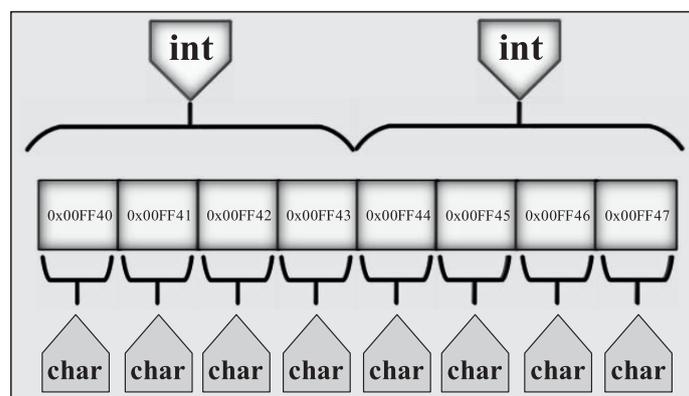


Рис. 1. Схема процесса обработки области памяти указателями разных типов
Fig. 1. Scheme for processing a memory area by pointers of different types

⁵ Статистика по СМДО [Электронный ресурс]. Режим доступа: https://nces.by/smdo_statistics/. Дата доступа: 26.01.2023.

⁶ Об информации, информатизации и защите информации [Электронный ресурс]. Режим доступа: <https://etalonline.by/document/?regnum=h10800455>. Дата доступа: 26.01.2023.

⁷ Современные языки программирования [Электронный ресурс]. Режим доступа: https://libeldoc.bsuir.by/bitstream/123456789/46982/1/Sovremennie_yaziki_programmirovaniya.pdf. Дата доступа: 26.01.2023.

Тем самым представлять каждый символ не в виде десятичного числа символа в кодировке ASCII, а в двоичной беззнаковой форме. Процесс обратного преобразования из числовых данных в текстовую информацию путем приведения типов позволяет придать набору цифр (чисел) читабельный вид.

Представим фрагмент кода, соответствующий алгоритму преобразования текстовой информации в числовые данные с помощью указателей:

```
...
1  for (int i = 0, j = 0; i < n; ++i, j += 4)
2  {
3      p = (uint*)(word + j);
4      fprintf(f, «%u\n», *p);
5  }
...
```

В приведенном фрагменте кода использованы следующие переменные: i – счетчик итераций; j – счетчик смещения указателя; n – количество итераций; p – указатель на область памяти целого беззнакового типа; $word$ – беззнаковый массив символов; f – указатель на файл; $uint$ – тип данных беззнаковый целый.

В ходе компиляции представленного фрагмента кода реализуется следующее:

- 1) ввод текста в ОЗУ с помощью клавиатуры;
- 2) расчет количества итераций явного приведения типов из `unsigned char*` в `unsigned int*` путем подсчета количества символов исходного текста без учета нуля терминатора (если количество символов не кратно 4, то количество итераций равно количеству символов, деленному на 4 + 1, иначе – деленному на 4);
- 3) операция явного приведения типов указателя на массив символов к указателю целого типа;
- 4) запись полученного значения в файл;
- 5) увеличение счетчика итераций на единицу;
- 6) увеличение счетчика на начало строки на 4;
- 7) повторение шагов 3, 4 до достижения количества итераций, полученных на шаге 2.

Схема алгоритма преобразования текстовой информации в числовые данные с помощью указателей представлена на рис. 2, схема алгоритма процесса обратного преобразования информации из числовой в текстовую с помощью указателей – на рис. 3.

Представим фрагмент кода, соответствующий алгоритму преобразования числовых данных в текстовую информацию с помощью указателей:

```
...
1  for (int i = 0; i < n; ++i)
2  {
3      fscanf_s(f, «%u», bp);
4      buf = (uchar*)bp;
5      for (int j = 0; j < 4; j++)
6          fprintf(f2, «%c», *(buf + j));
7  }
...
```

В приведенном фрагменте кода использованы следующие переменные: i – счетчик итераций; j – счетчик смещения указателя; n – количество итераций; buf – указатель на область памяти беззнакового символьного типа; bp – указатель беззнакового целого типа; $f, f2$ – указатели на файл; $uchar$ – тип данных беззнаковый символьный.

В ходе компиляции представленного фрагмента кода реализуется следующее:

- 1) считывание числовых данных из файла до тех пор, пока счетчик не примет значения количества итераций, полученных при преобразовании из текстовой информации в числовые данные;
- 2) операция явного приведения типов указателя целого типа к указателю на массив символов;
- 3) посимвольная запись полученных символов в файл;
- 4) повторение шагов 2, 3 до достижения количества итераций, полученных на шаге 1.

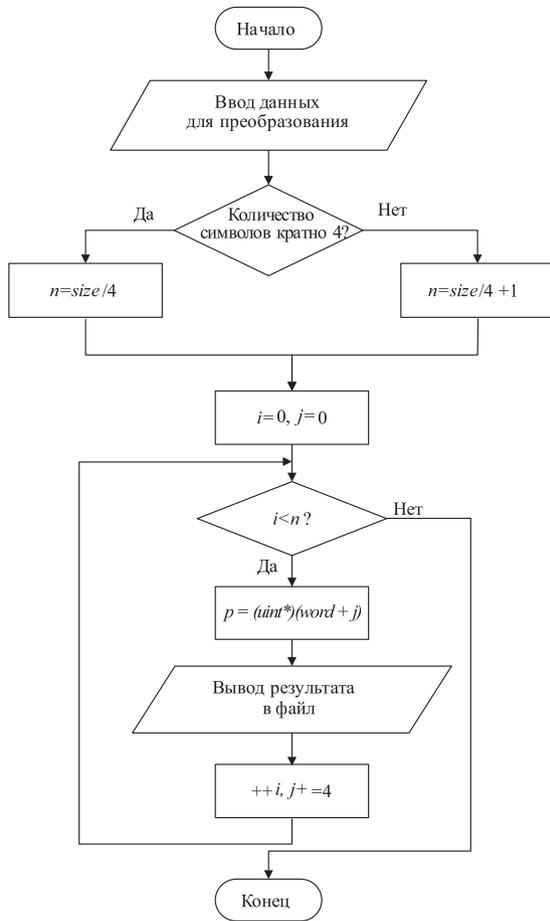


Рис. 2. Блок-схема алгоритма преобразования текстовой информации в числовые данные с помощью указателей

Fig. 2. Block diagram of the algorithm for converting text information into numeric data using pointers

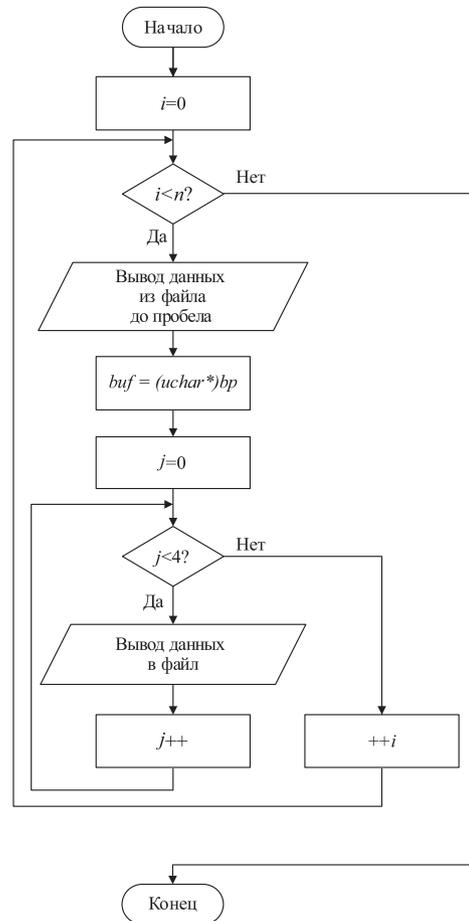


Рис. 3. Блок-схема алгоритма преобразования числовых данных в текстовую информацию с помощью указателей

Fig. 3. Block diagram of the algorithm for converting numeric data into text information using pointers

Результаты исследований и их обсуждение

Примеры работы проекта программного средства представлены на рис. 4, 5.

Для понимания того, как происходит преобразование, рассмотрим следующий пример. Представим слово Мата в виде числовых кодов таблицы ASCII. Это будут коды 77 97 109 97. С помощью операции присваивания и явного приведения типов происходит посимвольное преобразование из десятичного представления в двоичное (см. строку 3 фрагмента кода преобразования текстовой информации в числовые данные с помощью указателей):

$$\begin{aligned} 77_{10} &= 0100\ 1101_2; \\ 97_{10} &= 0110\ 0001_2; \\ 109_{10} &= 0110\ 1101_2; \\ 97_{10} &= 0110\ 0001_2. \end{aligned}$$

Происходит явное преобразование беззнакового символьного типа к беззнаковому числовому типу, поэтому порядок следования байт изменяется на противоположный, так как при двоичном представлении данных старший разряд стоит левее⁸, поскольку происходит оперирование не битами, а байтами. Таким образом, строка будет представлена в виде последовательности кодов 97 109 97 77. Все коды в двоичном виде конкатенируются и преобразуются в десятичное число. Используя калькулятор, найдем десятичное число, которое будет записано в файл (рис. 6).

⁸ Big-Endian, Little-Endian. Available: <https://learn.microsoft.com/en-us/style-guide/a-z-word-list-term-collections/b/big-endian-little-endian> (Accessed 26 January 2023).

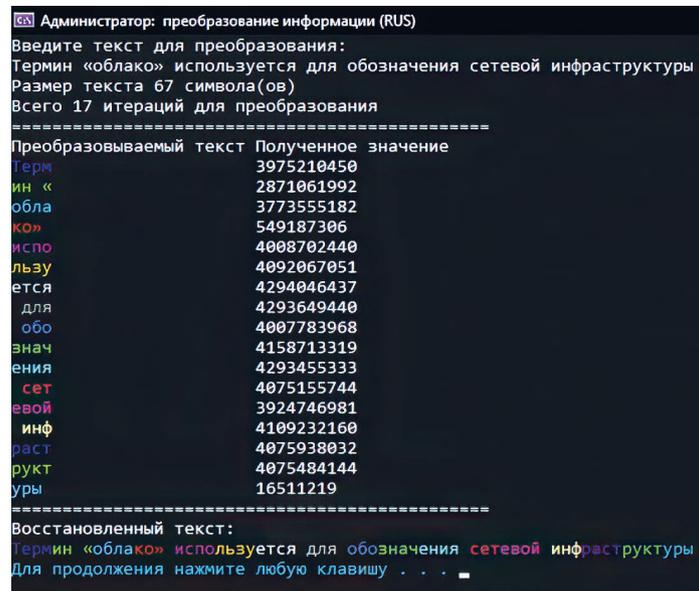


Рис. 4. Фрагмент диалогового окна с результатами преобразования текстовой информации на русском языке в числовые данные

Fig. 4. Dialog window fragment with the results of converting textual information in Russian into numerical data

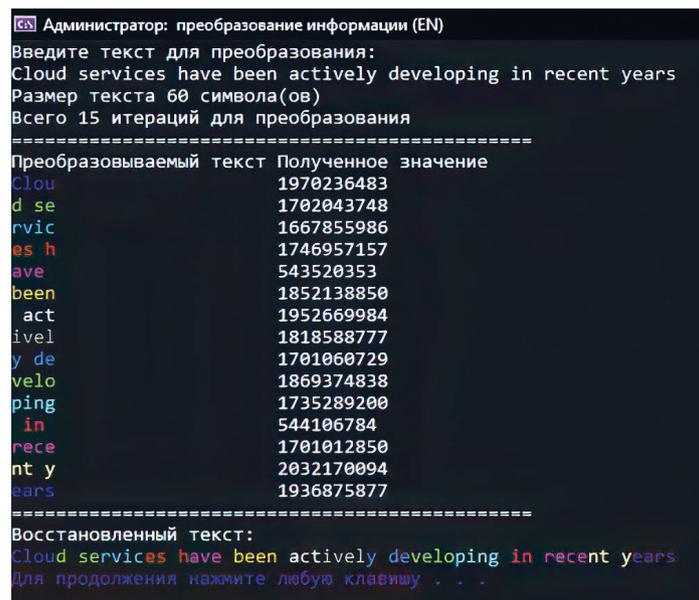


Рис. 5. Фрагмент диалогового окна с результатами преобразования текстовой информации на английском языке в числовые данные

Fig. 5. Dialog window fragment with the results for converting textual information in English into numeric data

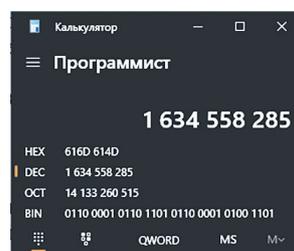


Рис. 6. Представление полученного числа из двоичного вида в десятичный

Fig. 6. Representing the resulting number from binary to decimal

При обратном преобразовании используется тот же алгоритм, который применялся для преобразования текстовой информации в числовые данные, только в обратном порядке. Число 1 634 558 285 с помощью операции приведения типов (см. строку 4 фрагмента кода преобразования числовых данных в текстовую информацию с помощью указателей) приведем к двоичному виду. В результате получим 8 групп бит по 4 бита в каждой группе:

0110 0001 0110 1101 0110 0001 0100 1101

Преобразуем группы бит в байты, то есть 1 байт будет состоять из 2 групп по 4 символа, или из 8 символов. Переведем полученные байты данных из двоичной системы счисления в десятичную:

$$\begin{aligned} 0110\ 0001_2 &= 1 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5 + 1 \cdot 2^6 + 0 \cdot 2^7 = 97_{10}; \\ 0110\ 1101_2 &= 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5 + 1 \cdot 2^6 + 0 \cdot 2^7 = 109_{10}; \\ 0110\ 0001_2 &= 1 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5 + 1 \cdot 2^6 + 0 \cdot 2^7 = 97_{10}; \\ 0100\ 1101_2 &= 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 0 \cdot 2^5 + 1 \cdot 2^6 + 0 \cdot 2^7 = 77_{10}. \end{aligned}$$

Порядок следования символов после преобразования изменился на противоположный. Это связано с тем же явлением, что наблюдалось при первичном преобразовании. При явном приведении типов происходит перевод десятичного значения в двоичное. Далее следует разделение двоичного значения на отдельные байты, перевод в десятичное представление и размещение байт по указателю в том порядке, который был до преобразования текстовой информации в числовые данные. Преобразование информации из примера представлено на рис. 7.

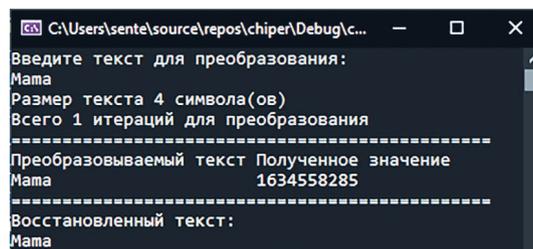


Рис. 7. Фрагмент окна программы с результатами преобразования информации в соответствии с примером
Fig. 7. Fragment of the program window with the results of information conversion according to the example

Программное средство, разработанное в соответствии с предложенным алгоритмом, может быть использовано при проведении лабораторных или практических занятий со студентами 1–2 курсов учреждений высшего образования. В табл. 1 представлены сведения о путях использования этого программного средства по предложенному выше назначению.

Таблица 1. Сведения о путях использования проекта программного средства при проведении занятий по различным дисциплинам

Table 1. Information about the ways of using the software project in conducting classes in various disciplines

Наименование дисциплин (-ы) / Name of discipline (-s)	Наименование тем (-ы) / Topic name (-s)	Путь использования программного средства при проведении лабораторных или практических занятий / The way of using the software during laboratory or practical classes
«Информационные системы и технологии»	Информация и данные. Виды и свойства информации; цели и виды преобразования информации; кодирование информации; шифрование данных (при условии, если программное средство будет усовершенствовано путем реализации в нем криптографических алгоритмов)	Использовать как наглядный пример процесса преобразования информации из одного представления в другое в ходе чтения лекций по указанным темам

Окончание табл. 1
Ending of Tab. 1

Наименование дисциплин (-ы) / Name of discipline (-s)	Наименование тем (-ы) / Topic name (-s)	Путь использования программного средства при проведении лабораторных или практических занятий / The way of using the software during laboratory or practical classes
«Основы алгоритмизации и программирования», «Конструирование программного обеспечения»	Операции отношения, логические операции; операции приведения типов; арифметические операции и их особенности по сравнению с другими языками программирования; неявное преобразование типов; операция преобразования типов; операторы цикла; указатели и операции над ними; связь между указателями и массивами. Организация работы с динамическими массивами; нуль-терминальные строки; функции для работы с нуль-терминальными строками. Алгоритмы работы с нуль-терминальными строками; текстовые файлы. Библиотечные функции для работы с файлами. Алгоритмы работы с файлами	1. Дать студентам задание по реализации фрагмента кода для восстановления текстовой информации из числовых данных, или наоборот, для более глубокого понимания процессов явного приведения типов, операций взятия адреса и разыменования. 2. Дать студентам задание по усовершенствованию программного средства путем реализации в нем криптографических алгоритмов
«Основы теории информации»	Понятие системы счисления. Позиционные и непозиционные системы счисления. Основание системы счисления; правила перевода из одной системы счисления в другую. Правила десятичной арифметики; способы кодирования и декодирования символьной информации. Представление символьной информации в ПК	Использовать как наглядный пример процесса преобразования информации из одного представления в другое в ходе чтения лекций по указанным темам

Заключение

1. Программное средство, разработанное в соответствии с созданным алгоритмом преобразования текстовой информации в числовые данные с помощью указателей, может быть использовано при проведении лабораторных и практических работ не только аудиторно, но и дистанционно. Для этого потребуется разместить исполняемый файл, исходный код, а также материалы, которые требуются для демонстрации, в системе электронного обучения.

2. Дальнейшие исследования будут направлены на усовершенствование предложенного алгоритма. В частности, в программном средстве, разработанном в соответствии с созданным алгоритмом, могут быть реализованы следующие возможности:

- считывание текста из файла, подлежащего преобразованию;
- разработка графического интерфейса пользователя с использованием интерактивных элементов для возможности пошаговой демонстрации выполнения действий преобразования на настольных и мобильных операционных системах;
- вывод преобразовываемого текста в двоичной системе счисления;
- поддержка логической операции «исключающее или» в целях обеспечения возможности внесения информационной энтропии;
- указание в явном виде прямого или обратного порядка байтов в программе (Big-endian, Little-endian);
- поддержка простейших криптографических операций.

3. Благодаря реализации в программном средстве представленных возможностей можно будет увеличить количество учебных дисциплин, при преподавании которых целесообразно его применение.

Список литературы

1. Носевич, В. Л. Плюсы и минусы электронного документооборота / В. Л. Носевич // Архівы і справаводства. 2013. Т. 3, № 87. С. 42–56.
2. Беляцкая, Т. Н. Готовность населения к экономическому поведению в условиях электронной экономики: проблемы электронного здравоохранения / Т. Н. Беляцкая, О. М. Маклакова // Цифровая трансформация. 2019. № 2. С. 13–28.
3. Головенчик, Г. Г. Современные тенденции цифрового реформирования образования / Г. Г. Головенчик // Цифровая трансформация. 2020. № 4. С. 5–20.
4. Лукьяненко, А. Ю. Оптимизация процесса выбора программного обеспечения в организации / А. Ю. Лукьяненко // Цифровая трансформация. 2019. № 4. С. 12–22.
5. Марков, А. Н. Готовность учреждений высшего образования к цифровой трансформации процессов / А. Н. Марков, С. А. Мигалевич // Цифровая трансформация. 2021. № 2. С. 64–68.
6. Седун, А. М. Опыт дистанционного обучения в Белорусском государственном экономическом университете / А. М. Седун, Е. А. Гриневич, А. И. Верещако // Цифровая трансформация. 2018. № 4. С. 56–60.
7. Старовойтова, Т. Ф. Цифровая трансформация системы сбора и обработки данных переписи населения Республики Беларусь / Т. Ф. Старовойтова, Т. И. Савченко // Цифровая трансформация. 2019. № 4. С. 39–49.

References

1. Nosevich V. L. (2013) Pros and Cons of Electronic Document Management. *Arhivy i Spravavodstva*. 3 (87), 42–56 (in Russian).
2. Byalyatskaya T. N., Maklakova O. M. (2019) Readiness of the Population for Economic Behavior in the Conditions of Electronic Economy: Problems of Electronic Health Care. *Digital Transformation*. (2), 13–28 (in Russian).
3. Golovenchik G. G. (2020) Modern Trends in Digital Education Reform. *Digital Transformation*. (4), 5–20 (in Russian).
4. Lukyanenko A. Yu. (2019) Optimization of the Software Selection Process in the Organization. *Digital Transformation*. (4), 12–22 (in Russian).
5. Markov A. N., Migalevich S. A. (2021) Readiness of Higher Education Institutions for Digital Transformation of Processes. *Digital Transformation*. (2), 64–68 (in Russian).
6. Sedun A. M., Grinevich E. A., Vereshchako A. I. (2018) The Experience of Distance Learning at the Belarusian State University of Economics. *Digital Transformation*. (4), 56–60 (in Russian).
7. Starovoitova T. F., Savchenko T. I. (2019) Digital Transformation of the System of Collection and Processing of Population Census Data of the Republic of Belarus. *Digital Transformation*. (4), 39–49 (in Russian).

Вклад авторов / Authors' contribution

Авторы внесли равный вклад в написание статьи / The authors contributed equally to the writing of the article.

Сведения об авторах

Герасимов В. А., сотрудник Научно-исследовательского института технической защиты информации, магистрант Белорусского государственного университета информатики и радиоэлектроники

Бойправ О. В., к. т. н., доцент, доцент кафедры защиты информации Белорусского государственного университета информатики и радиоэлектроники

Адрес для корреспонденции

220088, Республика Беларусь,
г. Минск, ул. Первомайская, 26, корп. 2
Научно-исследовательский институт
технической защиты информации
Тел.: +375 17 294-01-71
E-mail: v.gerasimov@bsuir.by
Герасимов Вячеслав Александрович

Information about the authors

Gerasimov V. A., Employee of the Scientific Research Institute of Technical Protection of Information, Master's Student at the Belarusian State University of Informatics and Radioelectronics

Boyprav O. V., Cand. of Sci., Associate Professor, Associate Professor at the Information Security Department of the Belarusian State University of Informatics and Radioelectronics

Address for correspondence

220088, Republic of Belarus,
Minsk, Pervomayskaya St., 26, build. 2
Scientific Research Institute
of Technical Protection of Information
Tel.: +375 17 294-01-71
E-mail: v.gerasimov@bsuir.by
Gerasimov Vyacheslav Alexandrovich