Лабораторная работа № 6

**Кодировки ASCII и Unicode**

Цель работы: формирование практических навыков обоснованного выбора подходящих форматов хранения текстовых данных исходя из требований технического задания.

Задачи: Кодирование символов текста целыми числами с использованием кодовых таблиц (ASCII, КОИ-7, КОИ-8) и ытандартаUnicode.

**Содержание отчета**

1. Титульный лист;
2. Цели и задачи лабораторной работы;
3. Условие задания (с данными варианта);
4. Подробное решение каждого задания;
5. Вывод.

Теоретическая часть

При представлении текста каждому символу алфавита ставится в соответствии некоторая последовательность бит (bit pattern) – **двоичный код.** Перечень символов и их двоичных кодов называется **кодировкой** (encoding).

8-битные кодировки

Тексты вводятся в память компьютера с помощью клавиатуры. В оперативную память они попадают в двоичном коде. Это значит, что каждый символ представляется 8-разрядным двоичным кодом.

Кодирование заключается в том, что каждому символу ставится в соответствие уникальный десятичный код от 0 до 255 или соответствующий ему двоичный код от 00000000 до 11111111. Таким образом, человек различает символы по их начертанию, а компьютер - по их коду.

Удобство побайтового кодирования символов очевидно, поскольку байт - наименьшая адресуемая часть памяти и, следовательно, процессор может обратиться к каждому символу отдельно, выполняя обработку текста. С другой стороны, 256 символов – это вполне достаточное количество для представления самой разнообразной символьной информации.

Понятно, что это дело условное, можно придумать множество способов кодировки.

Следует знать:

1. Все символы компьютерного алфавита пронумерованы от 0 до 255.
2. Каждому номеру соответствует восьмиразрядный двоичный код от 00000000 до 11111111.
3. Этот код просто порядковый номер символа в двоичной системе счисления.

Присвоение символу конкретного кода - это соглашение, которое фиксируется в кодовой таблице.

Кодировка ASCII-1967

Данная кодировка использует 7-битные кодовые последовательности (0016–7F16) и определяет коды для следующих символов (см. табл. Приложение 1):

* латинских букв;
* цифр;
* знаков препинания;
* некоторых математических символов;
* некоторых специальных символов;
* т. н. управляющих символов (например, CR и LF – символы возврата каретки и перевода строки; HT – горизонтальная табуляция, SP – пробел, ESC – альтернативный регистр, DEL – удаление, BEL – звуковой сигнал).

*Замечание*: в кодировке ASCII коды латинских букв образуют непрерывную возрастающую последовательность: «A» – 4116, «B» – 4216, …, «Z» – 5A16; аналогично и для строчных букв: «a» – 6116, «b» – 6216, …, «z» – 7A16. Непрерывную возрастающую последовательность образуют и коды цифр: «0» – 3016, «1» – 3116, …, «9» – 3916.

В дальнейшем ASCII стала международным стандартом под именем ISO/IEC 646:1991. В этом стандарте часть символов не определены и должны быть заданы в национальных реализациях (в [табл\_1](#Таблица_1). Приложение 1 выделены серым цветом).

8‑битные кодировки каждый символ текста кодируют 8‑битным целым числом. Для кодирования управляющих символов, символов латинского алфавита, цифр, знаков препинания и некоторых других символов используется таблица ASCII.

Существуют различные варианты дополнения кодировки ASCII до 8 битной.

Семейство стандартов ISO 8859 используется в операционных системах Unix. Стандарт ISO 8859 5 (см. [табл\_2.](#Таблица_2) Приложение 1) этого семейства определяет символы кириллицы и других славянских языков.

В операционных системах MS DOS и Windows различные варианты наборов символов для 8-битных кодов 8016–FF16 называют кодовыми страницами (code page). В MS DOS помимо символов национального алфавита кодовая страница содержала символы псевдографики, использовавшиеся для отображения графического интерфейса в текстовом режиме. С выходом Windows необходимость в псевдографике отпала, поэтому стали использоваться новые наборы кодовых страниц.

Кодовые страницы, содержащие символы кириллицы: MS DOS cp866 ([см. табл\_3.](#Таблица_3) Приложение 1) и Windows-1251 (см. [табл\_4.](#Таблица_4) Приложение 1).

**Задача 1.** Покажите, как текст «**Я выучил C++20!**» представляется с использованием 8‑битных кодировок:

а) cp866;

б) Windows‑1251;

в) ISO 8859‑5. **В ответе приведите шестнадцатеричные коды символов.**

***Решение задания «а»***

Найдем код для каждого символа текста. Коды русских букв будем искать по табл. 5.3 для «верхней» части кодовой страницы cp866, коды латинских букв («C»), знаков (пробел, «+», «!») и цифр – по [табл.1](#Таблица_1) для «нижней» части таблицы ASCII.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Символ** | **Я** | **[SP]** | **в** | **ы** | **у** | **ч** | **и** | **л** | **[SP]** | **C** | **+** | **+** | **2** | **0** | **!** |
| **Табл.** | 3 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Код** | 9F | 20 | A2 | EB | E3 | E7 | A8 | AB | 20 | 43 | 2B | 2B | 32 | 30 | 21 |

Т. о., символы заданного текста кодируется следующими числами: 9F 20 A2 EB E3 E7 A8 AB 20 43 2B 2B 32 30 21. Длина строки – 15 символов, длина кода – 15 байт.

*Ответ*: 9F 20 A2 EB E3 E7 A8 AB 20 43 2B 2B 32 30 21.

***Решение задания «б»***

Найдем код для каждого символа текста. Коды русских букв будем искать по [табл. 4](#Таблица_4) для «верхней» части кодовой страницы Windows‑1251, коды латинских букв («C»), знаков (пробел, «+», «!») и цифр – по [табл. 1](#Таблица_1) для «нижней» части таблицы ASCII.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Символ** | Я | [SP] | в | ы | у | ч | л | л | [SP] | C | + | + | 2 | 0 | ! |
| **Табл.** | 3 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Код** | DF | 20 | E2 | FB | F3 | F7 | E8 | EB | 20 | 43 | 2B | 2B | 32 | 30 | 21 |

Т. о., символы заданного текста кодируется следующими числами: DF 20 E2 FB F3 F7 E8 EB 20 43 2B 2B 32 30 21. Длина строки – 15 символов, длина кода – 15 байт.

*Ответ*: DF 20 E2 FB F3 F7 E8 EB 20 43 2B 2B 32 30 21.

***Решение задания «в»***

Найдем код для каждого символа текста. Коды русских букв будем искать по [табл.2](#Таблица_2) для «верхней» части кодировки ISO 8859‑5, коды латинских букв («C»), знаков (пробел, «+», «!») и цифр – по [табл.1](#Таблица_1) для «нижней» части таблицы ASCII.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Символ** | Я | [SP] | в | ы | у | ч | и | л | [SP] | C | + | + | 2 | 0 | ! |
| **Табл.** | 3 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Код** | CF | 20 | D2 | EB | E3 | E7 | D8 | DB | 20 | 43 | 2B | 2B | 32 | 30 | 21 |

Т. о., символы заданного текста кодируется следующими числами: CF 20 D2 EB E3 E7 D8 DB 20 43 2B 2B 32 30 21. Длина строки – 15 символов, длина кода – 15 байт.

*Ответ*: CF 20 D2 EB E3 E7 D8 DB 20 43 2B 2B 32 30 21.

**Задача 2.** Некоторый текст с использованием кодировки cp866 представляется в виде следующей последовательности кодов: 88 93 8A 34. Укажите текст, соответствующий приведенной последовательности.

***Решение задания***

Для каждого кода определим соответствующий символ. Коды 0016–7F16 будем искать по [табл.1](#Таблица_1) (общие коды ASCII), коды 8016–FF16 – по [табл.3](#Таблица_3) (кодовая страница cp866).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Код** | **88** | **93** | **8A** | **34** |
| **Таблица** | 3 | 3 | 3 | 1 |
| **Символ** | И | У | К | 4 |

Т. о., заданные коды определяют текст «**ИУК4**».

*Ответ*: «**ИУК4**».

Для записи текстовой (знаковой) информации всегда используется какой-либо язык (естественный или формальный).

**Вспомним**, что все множество используемых в языке символов называется **алфавитом.** Полное число символов алфавита **N** называют его **мощностью**. При записи текста в каждой очередной позиции может появиться любой из N символов алфавита, т.е. может произойти N событий.

Следовательно:

* каждый символ алфавита содержит i бит информации, где i определяется из неравенства (формула Хартли): 2 i ≥ N.
* общее количество информации в тексте определяется формулой: **V = k \* i,**

где **V** – количество информации в тексте;

**k** – число знаков в тексте (включая знаки препинания и даже пробелы),

**i** - количество бит, выделенных на кодирование одного знака.

**Задача 3.** Какой объем памяти займет заданный текст *«**Теоретическая информатика»,* если для его представления использовать кодировку Windows-1251?

***Решение*** Длина текста: 25 символов.

В 8 битных кодировках, к которым относится и Windows 1251, каждый символ текста кодируется одним байтом. Следовательно, объем кода текста: 25 Б.

Ответ: 25 Б

**Unicode**

В настоящее время для кодирования, представления и обработки текста используется стандарт **Unicode.** Этот стандарт каждому символу (*code point*) ставит в соответствие **21‑битное числовое** значение из диапазона 00000016–10FFFF16.

В пространстве Unicode можно выделить *блоки* (blocks) символов – непрерывные последовательности символов, относящиеся к одной «письменности». Например, блок «Controls and Basic Latin» с кодами 000016–007F16 ([табл.5](#Таблица_5)) эквивалентен таблице ASCII (ср. с [табл.1](#Таблица_1)), а блок «Cyrillic» с кодами 040016–04FF16 ([табл.6](#Таблица_6)) содержит символы кириллицы.

Символы Unicode будем записывать в виде «U+*xxxx*», где *xxxx* – шестнадцатеричное значение кода символа.

Существуют различные варианты представления 21‑битных кодов Unicode в памяти (в оперативной или во внешней – файлах на диске).

Рассмотрим следующие: UTF‑8, UTF‑16, UTF‑32.

**Представление UTF‑16**

Каждый код Unicode представляется в виде 2 байт или 4 байт.

Символы от U+0000 до U+D7FF и от U+E000 до U+FFFF кодируются 2 байтными значениями («как есть», т. е. код равен численному значению code point).

Символы от U+D800 до U+DFFF – зарезервированы (см. ниже), они не должны соответствовать реальным символам.

Символы от U+010000 до U+10FFFF кодируются 4 байтными значениями по следующему алгоритму.

*Алгоритм 1. Получение кода UTF‑16 для символов от U+010000 до U+10FFFF*

1. Из 21‑битного кода символа вычитается значение 1000016. В результате получается 20‑битное значение от 016 до FFFFF16.
2. К старшим 10 битам (значению от 016 до 3FF16) результата, полученного в п. 1, прибавляется значение D80016. Полученное 16‑битное значение (из диапазона D80016–DBFF16) становится первой половиной кода. Оно называется старшим суррогатом (high surrogate или leading surrogate).
3. К младшим 10 битам (значению от 016 до 3FF16) результата, полученного в п. 1, прибавляется значение DС0016. Полученное 16‑битное значение (из диапазона DC0016–DFFF16) становится второй половиной кода. Оно называется *младшим суррогатом* (low surrogate или trailing surrogate).
4. 2‑байтные значения, полученные в пунктах 2 и 3, образуют 4‑байтный код символа.

В зависимости от архитектуры аппаратного и программного обеспечения, UTF‑16 может записывать байты кода в разном порядке: Big-endian или Little-endian; соответствующие варианты представления принято обозначать UTF‑16BE и UTF‑16LE.

При сохранении текста в представлении UTF‑16 в файл, в начало файла может добавляться т. н. *маркер порядка байт* (byte order mark, BOM) –символ U+FEFF «ZERO WIDTH NON-BREAKING SPACE». Этот символ позволяет определить используемый порядок байт: если в начала файла находятся байты FE FF, то при сохранении использовался UTF‑16BE, если же байты FF FE, то использовался UTF‑16LE. Если BOM отсутствует, то стандарт предполагает использование UTF‑16BE (однако многие Windows-приложения предполагают UTF‑16LE, т. к. данная кодировка используется в операционной системе по умолчанию).

**Представление UTF‑32**

Каждый код Unicode представляется в виде 4 байт, содержащих числовое значение символа.

При хранении и передаче информации в данном представлении, также может использоваться маркер порядка байт, причем правила его использования аналогичны UTF‑16. Так, если первые байты: 00 00 FE FF, то используется UTF‑32BE. Если же первые байты FF FE 00 00, то используется UTF‑32LE.

**Представление UTF‑8**

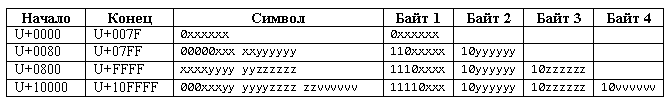
Каждый код Unicode представляется в виде 1, 2, 3 или 4 байт.

*Алгоритм 2. Получение кода UTF‑8 (см. также табл. 7)*

1. Для символа из диапазона от U+0000 до U+007F (соответствующего 7‑битной кодировке ASCII) используется 1‑байтный код, содержащий числовое значение символа.
2. Для символа из диапазона от U+0080 до U+07FF (в этот диапазон попадают и символы блока Cyrillic) используется 2‑байтный код. Числовое значение из рассматриваемого диапазона является 11‑битным. Запишем его следующим образом: 00000xxx xxyyyyyy2. Тогда 2 байта кода будут иметь следующий вид: 110xxxxx 10yyyyyy2.
3. Для символа из диапазона от U+0800 до U+FFFF используется 3‑байтный код. Числовое значение из рассматриваемого диапазона является 16‑битным. Запишем его следующим образом: xxxxyyyy yyzzzzzz2. Тогда 3 байта кода будут иметь следующий вид: 1110xxxx 10yyyyyy 10zzzzzz2.
4. Для символа из диапазона от U+10000 до U+10FFFF используется 4‑байтный код. Числовое значение из рассматриваемого диапазона является 21‑битным. Запишем его следующим образом: 000xxxyy yyyyzzzz zzvvvvvv2. Тогда 4 байта кода будут иметь следующий вид: 11110xxx 10yyyyyy 10zzzzzz 10vvvvvv2.
5. Первый байт последовательности называется leading byte, а последующие (в случае многобайтного кода) – continuation bytes.

*Замечание*. UTF‑8 представляет собой префиксный код: байты с префиксом «0» являются 1‑байтными кодами; байты с префиксом «110» – первые байты в 2‑байтном коде; байты с префиксом «1110» – первые байты в 3‑байтном коде; байты с префиксом «11110» – первые байты в 4‑байтном коды; байты с префиксом «10» – байты-продолжения (второй и последующий байты) в многобайтном коде. Можно заметить, что количество единичных битов в начале первого байта равно общему количеству байт в коде (исключение – однобайтные коды).

**Таблица 7**. Кодировка UTF‑8



При хранении и передаче информации в представлении UTF‑8 байты всегда следуют в порядке, который определен [алгоритмом 2,](#Алгоритм_2) т. е. вариаций BE и LE для данного представления не существует. Однако, иногда может использоваться маркер порядка байт EF BB BF, обозначающий кодировку UTF‑8 (этот маркер является представлением символа U+FEFF в UTF‑8).

**Задача 4.** Покажите, какие символы Unicode соответствуют символам заданного текста:

«**Я выучил C++20!**».

Каждый код укажите в формате U+*xxxx*, где *xxxx* – шестнадцатеричное представление кода.

Покажите, как эти символы представляются с использованием UTF‑8, UTF‑16, UTF‑32.

Ответ приведите в виде шестнадцатеричных кодов, без маркера порядка байт (BOM).

***Решение***

Найдем символы текста в таблицах Unicode. Коды русских букв будем искать в блоке Cyrillic (символы 040016 – 04FF16, [табл.6](#Таблица_6)). Коды латинских букв, цифр и знаков будем искать в блоке Controls and Basic Latin (символы 000016 – 007F16, [табл.5](#Таблица_5)).

| **Символ** | **Таблица** | **Символ Unicode** |
| --- | --- | --- |
| Я | 6 | 042F |
| [SP] | 5 | 0020 |
| в | 6 | 0432 |
| ы | 6 | 044B |
| у | 6 | 0443 |
| ч | 6 | 0447 |
| и | 6 | 0438 |
| л | 6 | 043B |
| [SP] | 5 | 0020 |
| C | 5 | 0043 |
| + | 5 | 002B |
| + | 5 | 002B |
| 2 | 5 | 0032 |
| 0 | 5 | 0030 |
| ! | 5 | 0021 |

Т. о., символы заданного текста представляют собой следующие символы Unicode: U+042F U+0020 U+0432 U+044B U+0443 U+0447 U+0438 U+043B U+0020 U+0043 U+002B U+002B U+0032 U+0030 U+0021.

**Получим** **представление в UTF‑8**.

Символы U+0020, U+0043, U+002B, U+0032, U+0030 и U+0021 будут иметь 1‑байтные коды, т. к. они принадлежат диапазону от U+0000 до U+007F (см.п. 1 [алгоритма 2](#Алгоритм_2) и строку 1 [табл. 7](#Таблица_7)).

«Пробел (SP)»: U+0020 🡪 001000002 = 2016.

**«С»:** U+0043 🡪 010000112 = 4316.

**«+»**: U+002B 🡪 001010112 = 2B16.

**«2»:** U+0031 🡪 001100012 = 3216.

**«0»:** U+0038 🡪 001110002 = 3016.

**«!»:** U+0021 🡪 001000012 = 2116.

Остальные символы (в рассматриваемом примере – русские буквы) будут иметь 2‑байтные коды, т. к. они принадлежат диапазону от U+0080 до U+07FF.

Построим код UTF‑8 для символа U+042F (буква «Я»). Запишем численное значение символа в двоичном виде: 00000100 001011112. Оставим 11 значащих бит (см. п. 2 [алгоритма 2](#Алгоритм_2) и строку 2 [табл. 7](#Таблица_7)): 100001011112. Разобьем на группы из 5 бит и 6 бит: 10000-1011112.

Старший байт кода символа имеет вид: 110xxxxx2, а младший байт – 10yyyyyy2. xxxxx = 100002 (старшие 5 бит числового значения символа); yyyyyy = 1011112 (младшие 6 бит числового значения символа).

Т. о., код символа U+042F имеет вид: 11010000 101011112 = D0 AF16.

Аналогично получаем коды остальных русских букв.

**«в»:** U+0432 = 00000100001100102 = 100001100102 = 10000-1100102 🡪 11010000 101100102 = D0 B216.

**«ы»:** U+044B = 00000100010010112 = 100010010112 = 10001-0010112 🡪 11010001 100010112 = D1 8B16.

**«у»:** U+0443 = 00000100010000112 = 100010000112 = 10001-0000112 🡪 11010001 100000112 = D1 8316.

**«ч»:** U+0447 = 00000100010001112 = 100010001112 = 10001-0001112 🡪 11010001 100001112 = D1 8716.

**«и»:** U+0438 = 00000100001110002 = 100001110002 = 10000-1110002 🡪 11010000 101110002 = D0 B816.

**«л»:** U+043B = 00000100001110112 = 100001110112 = 10000-1110112 🡪 11010000 101110112 = D0 BB2.

Т. о., в UTF‑8 заданный текст имеет вид:

D0 AF 20 D0 B2 D1 8B D1 83 D1 87 D0 B8 D0 BB 20 43 2B 2B 31 38 21.

**Получим** **представление в UTF‑16**.

Все символы принадлежат диапазону от U+0000 до U+D7FF, поэтому они будут иметь 2‑байтные коды, совпадающие с числовым значением символа.

Т. о., в UTF‑16BE заданный текст имеет вид: 04 2F 00 20 04 32 04 4B 04 43 04 47 04 38 04 3B 00 20 00 43 00 2B 00 2B 00 32 00 30 00 21.

В UTF‑16LE заданный текст имеет вид: 2F 04 20 00 32 04 4B 04 43 04 47 04 38 04 3B 04 20 00 43 00 2B 00 2B 00 32 00 30 00 21 00.

**Получим** **представление в UTF‑32**.

Все символы будут иметь 4‑байтные коды, совпадающие с числовым значением символа.

Т. о., в UTF‑32 заданный текст имеет вид: 00 00 04 2F 00 00 00 20 00 00 04 32 00 00 04 4B 00 00 04 43 00 00 04 47 00 00 04 38 00 00 04 3B 00 00 00 20 00 00 00 43 00 00 00 2B 00 00 00 2B 00 00 00 31 00 00 00 34 00 00 00 21.

*Ответ*:

U+042F U+0020 U+0432 U+044B U+0443 U+0447 U+0438 U+043B U+0020 U+0043 U+002B U+002B U+0032 U+0030 U+0021 **(символы Unicode);**

D0 AF 20 D0 B2 D1 8B D1 83 D1 87 D0 B8 D0 BB 20 43 2B 2B 31 38 21 **(UTF‑8);**

04 2F 00 20 04 32 04 4B 04 43 04 47 04 38 04 3B 00 20 00 43 00 2B 00 2B 00 32 00 30 00 21 **(UTF‑16BE);**

2F 04 20 00 32 04 4B 04 43 04 47 04 38 04 3B 04 20 00 43 00 2B 00 2B 00 32 00 30 00 21 00 **(UTF‑16LE);**

00 00 04 2F 00 00 00 20 00 00 04 32 00 00 04 4B 00 00 04 43 00 00 04 47 00 00 04 38 00 00 04 3B 00 00 00 20 00 00 00 43 00 00 00 2B 00 00 00 2B 00 00 00 32 00 00 00 30 00 00 00 21 **(UTF‑32).**

**Задача 4.** Покажите, как в кодировках **UTF‑16LE и UTF‑8** представляется заданный текст:

U+10E6D U+20AC U+10000 U+10437 U+1D11E;

*Решение задания*

Получим представление текста в UTF‑16LE.

Символ U+20AC принадлежит диапазону от U+0000 до U+D7FF, поэтому он будет иметь 2‑байтный код, совпадающий с числовым значением символа: AC 20 (LE).

Остальные символы принадлежат диапазону от U+010000 до U+10FFFF, а значит кодируются 4‑байтными значениями.

Построим код UTF‑16 для символа U+10E6D. Выполним вычитание значения 1000016:

10E6D16 – 1000016 = 0E6D16.

Запишем результат в двоичном виде, дополнив до 20 бит, и разобьем на две группы по 10 бит:

0E6D16 = 000000001110011011012 = 0000000011-10011011012.

Старший суррогат в паре получаем, прибавляя к первой группе бит значение D80016 = 11011000000000002:

11011000000000002 + 00000000112 = 11011000000000112 = D80316.

Младший суррогат в паре получаем, прибавляя ко второй группе бит значение DC0016 = 11011100000000002:

11011100000000002 + 10011011012 = 11011110011011012 = DE6D16.

Аналогично строим коды UTF‑16 для остальных символов.

*U+10000*: 1000016 – 1000016 = 000016 = 000000000000000000002 = 0000000000-00000000002. Старший суррогат: 11011000000000002 + 00000000002 = 11011000000000002 = D80016. Младший суррогат: 11011100000000002 + 00000000002 = 1101110000000002 = DС0016.

*U+10437*: 1043716 – 1000016 = 043716 = 000000000100001101112 = 0000000001-00001101112. Старший суррогат: 11011000000000002 + 00000000012 = 11011000000000012 = D80116. Младший суррогат: 11011100000000002 + 00001101112 = 11011100001101112 = DС3716.

*U+1D11E*: 1D11E16 – 1000016 = D11E 16 = 000011010001000111102 = 0000110100-01000111102. Старший суррогат: 11011000000000002 + 00001101002 = 11011000001101002 = D83416. Младший суррогат: 11011100000000002 + 01000111102 = 11011101000111102 = DD1E16.

Итак, учитывая порядок байт Little-endian, получаем следующую последовательность кодов: 03 D8 6D DE AC 20 00 D8 00 DС 01 D8 37 DС 34 D8 1E DD.

Получим представление в UTF‑8.

Все символы, кроме U+20AC, будут иметь 4‑байтные коды, т. к. они принадлежат диапазону от U+10000 до U+10FFFF. Символ U+20AC принадлежит диапазону от U+0800 до U+FFFF, поэтому будет иметь 3‑байтный код.

Построим код UTF‑8 для символа U+20AC. Запишем численное значение символа в двоичном виде: 00100000 101011002. Разобьем на группы из 4 бит, 6 бит и 6 бит: 0010-000010-1011002.

Первый байт кода символа имеет вид: 1110xxxx2, второй байт – 10yyyyyy2, а младший байт – 10zzzzzz2. xxxx = 00102 (старшие 4 бит числового значения символа); yyyyyy = 0000102 (следующие 6 бит числового значения символа); zzzzzz = 1011002 (младшие 6 бит числового значения символа).

Т. о., код символа U+20AC имеет вид: 11100010 10000010 101011002 = E2 82 AC16.

Построим код UTF‑8 для символа U+10E6D. Запишем численное значение символа в двоичном виде: 00000001 00001110 011011012. Отбросим старшие 3 бита (всегда нули) и разобьем на группу из 3 бит и 3 группы из 6 бит: 000-010000-111001-1011012.

Первый байт кода символа имеет вид: 11110xxx2, второй байт – 10yyyyyy2, третий байт – 10zzzzzz2, а младший байт – 10vvvvvv2. xxx = 0002 (старшие 3 бит числового значения символа); yyyyyy = 0100002 (следующие 6 бит числового значения символа); zzzzzz = 1110012 (следующие 6 бит числового значения символа); vvvvvv = 1011012 (следующие 6 бит числового значения символа).

Т. о., код символа U+10E6D имеет вид: 11110000 10010000 10111001 101011012 = F0 90 B9 AD16.

Аналогично получаем коды остальных символов.

*U+10000*: 1000016 = 00000001 00000000 000000002 = 000-010000-000000-0000002 🡪 11110000 10010000 10000000 100000002 = F0 90 80 8016.

*U+10437*: 1043716 = 00000001 00000100 001101112 = 000-010000-010000-1101112 🡪 11110000 10010000 10010000 101101112 = F0 90 90 B716.

*U+1D11E*: 1D11E16 = 00000001 11010001 000111102 = 000-011101-000100-0111102 🡪 11110000 10011101 10000100 100111102 = F0 9D 84 9E16.

Итак, получаем следующую последовательность кодов: F0 90 B9 AD E2 82 AC F0 90 80 80 F0 90 90 B7 F0 9D 84 9E.

*Ответ*: 03 D8 6D DE AC 20 00 D8 00 DС 01 D8 37 DС 34 D8 1E DD (UTF‑16LE); F0 90 B9 AD E2 82 AC F0 90 80 80 F0 90 90 B7 F0 9D 84 9E (UTF‑8).

**Задача 4.** Текстовый файл содержит следующую последовательность байт: FF FE 26 04 51 04 40 04 2C 00 6D 00 64 00 62 00 67 00 69 00 6A 00 6D 00 2E 00;

Зная, что текст был сохранен с использованием представления UTF-16, определите последовательность кодов Unicode символов текста (каждый код укажите в формате U+*xxxx*, где *xxxx* – шестнадцатеричное представление кода). Укажите тест, соответствующий полученной последовательности кодов Unicode.

Каков будет размер текстового файла, если этот текст сохранить с использованием представления UTF-8 и записать в начало файла маркер порядка байт (BOM)?

*Решение*

Начало файла – байты FF FE – это маркер порядка байт для представления UTF‑16LE.

Символы текста представлены 2 или 4 байтами, идущими в обратном порядке.

| **Байты** | **Символ Unicode** | | **Таблица** | **Символ** |
| --- | --- | --- | --- | --- |
| 26 04 | U+0426 | 6 | | Ц |
| 51 04 | U+0451 | 6 | | ё |
| 40 04 | U+0440 | 6 | | р |
| 2C 00 | U+002C | 5 | | , |
| 6D 00 | U+006D | 5 | | m |
| 64 00 | U+0064 | 5 | | d |
| 62 00 | U+0062 | 5 | | b |
| 67 00 | U+0067 | 5 | | g |
| 69 00 | U+0069 | 5 | | i |
| 6A 00 | U+006A | 5 | | j |
| 6D 00 | U+006D | 5 | | m |
| 2E 00 | U+002E | 5 | | . |

Т. о., текстовый файл содержит следующий текст: «**Цёр,mdbgijm.**»

В представлении UTF‑8 коды русских букв имеют длину 2 байта (см. табл. 6 и 7), а коды латинских букв, цифр и знаков – длину 1 байт (см. табл. 5 и 7). В тексте: русских букв – 3, латинских букв – 7, цифр – 0, знаков – 2. Следовательно, общая длина текста составит 15 байт. С учетом 3‑байтного BOM размер текстового файла в представлении UTF‑8 будет 18 байт.

*Ответ*: U+0426 U+0451 U+0440 U+002C U+006D U+0064 U+0062 U+0067 U+0069 U+006A U+006D U+002E; текст «**Цёр,mdbgijm.**»; 18 байт в UTF‑8.

**Варианты**

[**Задача 1.**](#Задача_1)Покажите, как текст варианта представляется с использованием 8 битных кодировок: cp866, Windows‑1251; ISO 8859‑5.

В ответе приведите шестнадцатеричные коды символов.

**Варианты задания:**

|  |  |
| --- | --- |
| № варианта | Текст |
| 1 | DF#яажф7йщь2? |
| 2 | KL1:пхн$Дкрн~ |
| 3 | Ше;нкGFяуюх3~ |
| 4 | NY,дсюж5хотп! |
| 5 | Эя2@укл;уfsф~ |
| 6 | М3н;лноёзжks? |
| 7 | L$тоб3%хяqхф\* |
| 8 | Узи1:кгfuамо= |
| 9 | Vzу8:би$дщпа |
| 10 | Zоз:соцзпяиf= |
| 11 | Dд$ц6хщсоiуж+ |
| 12 | NРю,жмsтышя4= |
| 13 | S2щ:рцзjцеяс~ |
| 14 | Ием1$Ydйдлт3! |
| 15 | Л6:7фбпнsvфр~ |
| 16 | Фщп$нагgnгт4! |
| 17 | V9:совзпяиф\* |
| 18 | Сф9#охюDZисй/ |
| 19 | Kf7:пхндкун~ |
| 20 | Цяя:Лвшdvцх5= |
| 21 | Тя2:поkmнгт2? |
| 22 | М4#грбlsгоаф. |

[**Задача 2.**](#Задача_2)Некоторый текст с использованием кодировки cp866 представляется в виде следующей последовательности кодов - Таблица вариантов. Укажите текст, соответствующий приведенной последовательности, исправьте допущенные ошибки и приведите правильный текст. Какой объем памяти займет заданный текст, если для его представления использовать кодировку Windows-1251?

**Варианты задания:**

|  |  |
| --- | --- |
| № варианта | Текст |
| 1 | 31 20 AA A8 AB AE A 1A0 A9 E2 3D 31 30 32 34 83 A1 A0 A9 E2 A0 AC |
| 2 | 31 20 AC A5 A3 A0 A1 A0 A9 E2 20 3D 31 30 32 34 A1 A0 A9 E2 A0 |
| 3 | 31 20 A3 A8 A3 A0 A1 A0 A9 E2 20 3D 31 30 32 34 8C A1 A0 A9 E2 5A |
| 4 | 31 20 E2 A5 E0 A0 A1 A0 A9 E2 20 3D 31 30 32 34 8A A1 A0 A9 E2 5A |
| 5 | 31 20 AA A8 AB AE A1 A0 A9 E2 20 3D 31 30 32 34 8A A1 A0 A9 E2 5A |
| 6 | A9 AE E2 E2 A0 A1 A0 A9 E2 3D 31 30 32 34 A7 A5 A1 A8 A1 A0 A9 E2 A0 AC |
| 7 | 8F A5 E2 A0 A1 A0 A9 E2 20 3D 31 30 32 34 20 E2 A5 E0 A0 A1 A0 A9 E2 |
| 8 | 9D AA E1 A0 A1 A0 A9 E2 20 3D 31 30 32 34 20 AF A8 E2 A0 A1 A0 A9 E2 |
| 9 | A2 20 31 20 92 A1 A0 A9 E2 A5 20 31 32 34 30 20 83 A1 A0 A9 E2 |
| 10 | A2 20 AF EF E2 A8 20 A1 A0 A9 E2 A0 E5 20 32 35 36 20 A1 A8 E2 |
| 11 | E4 AB AE AF 2D 38 AF AE E1 AB A5 A4 AE A2 A0 E2 A5 AB ED AD EC E5 20 A1 A8 Е2 |
| 12 | E1 A8 AC A2 AE AB EC 20 AE E2 20 A4 AE 20 31 32 37 20 E6 A8 E4 E0 EC |
| 13 | A2 20 AE A4 AD AE AC 20 A1 A0 A9 E2 A5 20 31 30 20 A1 A8 E2 |
| 14 | 31 20 E2 A5 E0 A0 A1 A0 A9 E2 20 3D 31 30 32 34 8A A1 A0 A9 E2 5A |
| 15 | 31 20 AA A8 AB AE A1 A0 A9 E2 20 3D 31 30 32 34 8A A1 A0 A9 E2 5A |
| 16 | A9 AE E2 E2 A0 A1 A0 A9 E2 3D 31 30 32 34 A7 A5 A1 A8 A1 A0 A9 E2 A0 AC |
| 17 | 8F A5 E2 A0 A1 A0 A9 E2 20 3D 31 30 32 34 20 E2 A5 E0 A0 A1 A0 A9 E2 |
| 18 | 31 20 AC A5 A3 A0 A1 A0 A9 E2 20 3D 31 30 32 34 A1 A0 A9 E2 A0 |
| 19 | 31 20 A3 A8 A3 A0 A1 A0 A9 E2 20 3D 31 30 32 34 8C A1 A0 A9 E2 5A |
| 20 | A2 20 31 20 92 A1 A0 A9 E2 A5 20 31 32 34 30 20 83 A1 A0 A9 E2 |
| 21 | A2 20 AF EF E2 A8 20 A1 A0 A9 E2 A0 E5 20 32 35 36 20 A1 A8 E2 |

[Задача 3.](#Задача_3) Текстовый файл содержит следующую последовательность байт: - Таблица вариантов. Зная, что текст был сохранен с использованием представления UTF-8, определите последовательность кодов Unicode символов текста (каждый код укажите в формате U+xxxx, где xxxx – шестнадцатеричное представление кода). Укажите тест, соответствующий полученной последовательности кодов Unicode.

Каков будет размер текстового файла, если этот текст сохранить с использованием представления UTF-8 и записать в начало файла маркер порядка байт (BOM)?

**Варианты задания:**

|  |  |
| --- | --- |
| № варианта | Текст |
| 1 | D0 A2 D1 88 D0 B9 D0 B0 24 71 72 72 6D 7E |
| 2 | 56 66 76 73 25 D1 80 D0 B4 D1 81 D0 BF D1 83 21 |
| 3 | EF BB BF D0 92 D1 88 D0 BE D0 B3 3A 66 77 71 68 67 21 |
| 4 | EF BB BF 44 62 6D 73 2C D0 BF D1 80 D0 B9 D1 84 D0 B6 D1 8F 3D |
| 5 | 42 73 66 72 24 D0 A3 D0 BF D0 B2 D0 B0 21 |
| 6 | D0 A8 D1 89 D0 B7 D1 80 40 76 68 7A 72 67 66 3F |
| 7 | EF BB BF 4B 73 73 D0 BB 2C D0 93 D1 84 D0 B8 D1 8E 21 |
| 8 | D0 95 D0 B4 D1 8F 40 6B 6D 6A 6D 39 2E |
| 9 | EF BB BF 51 7A 66 64 2C D0 A5 D0 B7 D1 80 71 3D |
| 10 | EF BB BF D0 92 D1 8D D1 80 67 3A 6B 64 76 66 64 3F |
| 11 | 48 66 76 77 3A D0 B4 D0 B5 D0 B8 D0 B9 21 |
| 12 | EF BB BF 42 6D 72 72 2C D1 8F D0 BA D0 B0 D0 BC 21 |
| 13 | D0 9B D1 83 D1 91 D1 81 24 6A 74 6B 72 73 2E |
| 14 | EF BB BF 52 76 67 77 3B D0 9B D0 B4 D0 B7 D1 80 77 7E |
| 15 | EF BB BF D0 A8 D0 BB D0 BF D0 B7 40 6A 77 74 76 77 74 7E |
| 16 | 46 6B 62 64 3A D0 B8 D1 91 D0 B6 D0 BA 68 21 |
| 17 | 53 66 71 6B 40 D1 88 D1 80 D1 80 D1 82 7E |
| 18 | 56 64 62 D0 B9 25 D0 BB D0 B0 D1 82 D0 BF D1 8E D1 8F 21 |
| 19 | D0 9C D0 B0 D1 82 D0 B2 3A 6A 6A 73 67 62 68 3D |
| 20 | EF BB BF 42 6D 72 6D 23 D0 B8 D0 B8 D1 84 D1 91 7E |
| 21 | D0 AE D0 B3 D0 BB 66 2C 4D 62 68 D0 BE 7E |

**ПРИЛОЖЕНИЕ 1**

**Таблица 1**. Кодировка ASCII-1967

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **A** | **B** | **C** | **D** | **E** | **F** |
| **0** | NUL  00 | SOH  01 | STX  02 | ETX  03 | EOT  04 | ENQ  05 | ACK  06 | BEL  07 | BS  08 | HT  09 | LF  0A | VT  0B | FF  0C | CR  0D | SO  0E | SI  0F |
| **1** | DLE  10 | DC1  11 | DC2  12 | DC3  13 | DC4  14 | NAK  15 | SYN  16 | ETB  17 | CAN  18 | EM  19 | SUB  1A | ESC  1B | FS  1C | GS  1D | RS  1E | US  1F |
| **2** | SP  20 | !  21 | "  22 | #  23 | $  24 | %  25 | &  26 | '  27 | (  28 | )  29 | \*  2A | +  2B | ,  2C | -  2D | .  2E | /  2F |
| **3** | 0  30 | 1  31 | 2  32 | 3  33 | 4  34 | 5  35 | 6  36 | 7  37 | 8  38 | 9  39 | :  3A | ;  3B | <  3C | =  3D | >  3E | ?  3F |
| **4** | @  40 | A  41 | B  42 | C  43 | D  44 | E  45 | F  46 | G  47 | H  48 | I  49 | J  4A | K  4B | L  4C | M  4D | N  4E | O  4F |
| **5** | P  50 | Q  51 | R  52 | S  53 | T  54 | U  55 | V  56 | W  57 | X  58 | Y  59 | Z  5A | [  5B | \  5C | ]  5D | ^  5E | \_  5F |
| **6** | `  60 | a  61 | b  62 | c  63 | d  64 | e  65 | f  66 | g  67 | h  68 | i  69 | j  6A | k  6B | l  6C | m  6D | n  6E | o  6F |
| **7** | p  70 | q  71 | r  72 | s  73 | t  74 | u  75 | v  76 | w  77 | x  78 | y  79 | z  7A | {  7B | |  7C | }  7D | ~  7E | DEL  7F |

**Таблица 2. Кодировка ISO 8859-5 (Cyrillic)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **A** | **B** | **C** | **D** | **E** | **F** |
| **8** | PAD  80 | HOP  81 | BPH  82 | NBH  83 | IND  84 | NEL  85 | SSA  86 | ESA  87 | HTS  88 | HTJ  89 | VTS  8A | PLD  8B | PLU  8C | RI  8D | SS2  8E | SS3  8F |
| **9** | DCS  90 | PU1  91 | PU2  92 | STS  93 | CCH  94 | MW  95 | SPA  96 | EPA  97 | SOS  98 | SGCI  99 | SCI  9A | CSI  9B | ST  9C | OCS  9D | PM  9E | APC  9F |
| **A** | NBSP  A0 | Ё  A1 | Ђ  A2 | Ѓ  A3 | Є  A4 | Ѕ  A5 | І  A6 | Ї  A7 | Ј  A8 | Љ  A9 | Њ  AA | Ћ  AB | Ќ  AC | SHY  AD | Ў  AE | Џ  AF |
| **B** | А  B0 | Б  B1 | В  B2 | Г  B3 | Д  B4 | Е  B5 | Ж  B6 | З  B7 | И  B8 | Й  B9 | К  BA | Л  BB | М  BC | Н  BD | О  BE | П  BF |
| **C** | Р  C0 | С  C1 | Т  C2 | У  C3 | Ф  C4 | Х  C5 | Ц  C6 | Ч  C7 | Ш  C8 | Щ  C9 | Ъ  CA | Ы  CB | Ь  CC | Э  CD | Ю  CE | Я  CF |
| **D** | а  D0 | б  D1 | в  D2 | г  D3 | д  D4 | е  D5 | ж  D6 | з  D7 | и  D8 | й  D9 | к  DA | л  DB | м  DC | н  DD | о  DE | п  DF |
| **E** | р  E0 | с  E1 | т  E2 | у  E3 | ф  E4 | х  E5 | ц  E6 | ч  E7 | ш  E8 | щ  E9 | ъ  EA | ы  EB | ь  EC | э  ED | ю  EE | я  EF |
| **F** | №  F0 | ё  F1 | ђ  F2 | ѓ  F3 | є  F4 | ѕ  F5 | і  F6 | ї  F7 | ј  F8 | љ  F9 | њ  FA | ћ  FB | ќ  FC | §  FD | ў  FE | џ  FF |

**ПРИЛОЖЕНИЕ 1**

**Таблица 3. Кодовая страница cp866 (MS-DOS Cyrillic)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **A** | **B** | **C** | **D** | **E** | **F** |
| **8** | А  80 | Б  81 | В  82 | Г  83 | Д  84 | Е  85 | Ж  86 | З  87 | И  88 | Й  89 | К  8A | Л  8B | М  8C | Н  8D | О  8E | П  8F |
| **9** | Р  90 | С  91 | Т  92 | У  93 | Ф  94 | Х  95 | Ц  96 | Ч  97 | Ш  98 | Щ  99 | Ъ  9A | Ы  9B | Ь  9C | Э  9D | Ю  9E | Я  9F |
| **A** | а  A0 | б  A1 | в  A2 | г  A3 | д  A4 | е  A5 | ж  A6 | з  A7 | и  A8 | й  A9 | к  AA | л  AB | м  AC | н  AD | о  AE | п  AF |
| **B** | ░  B0 | ▒  B1 | ▓  B2 | │  B3 | ┤  B4 | ╡  B5 | ╢  B6 | ╖  B7 | ╕  B8 | ╣  B9 | ║  BA | ╗  BB | ╝  BC | ╜  BD | ╛  BE | ┐  BF |
| **C** | └  C0 | ┴  C1 | ┬  C2 | ├  C3 | ─  C4 | ┼  C5 | ╞  C6 | ╟  C7 | ╚  C8 | ╔  C9 | ╩  CA | ╦  CB | ╠  CC | ═  CD | ╬  CE | ╧  CF |
| **D** | ╨  D0 | ╤  D1 | ╥  D2 | ╙  D3 | ╘  D4 | ╒  D5 | ╓  D6 | ╫  D7 | ╪  D8 | ┘  D9 | ┌  DA | █  DB | ▄  DC | ▌  DD | ▐  DE | ▀  DF |
| **E** | р  E0 | с  E1 | т  E2 | у  E3 | ф  E4 | х  E5 | ц  E6 | ч  E7 | ш  E8 | щ  E9 | ъ  EA | ы  EB | ь  EC | э  ED | ю  EE | я  EF |
| **F** | Ё  F0 | ё  F1 | Є  F2 | є  F3 | Ї  F4 | ї  F5 | Ў  F6 | ў  F7 | °  F8 | ∙  F9 | ·  FA | √  FB | №  FC | ¤  FD | ■  FE | NBSP  FF |

Байт - Byte - восемь бит.

81 A0 A9 E2 4279

**Таблица 4. Кодовая страница Windows-1251 (cp1251)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **A** | **B** | **C** | **D** | **E** | **F** |
| **8** | Ђ  80 | Ѓ  81 | ‚  82 | ѓ  83 | „  84 | …  85 | †  86 | ‡  87 | €  88 | ‰  89 | Љ  8A | ‹  8B | Њ  8C | Ќ  8D | Ћ  8E | Џ  8F |
| **9** | ђ  90 | ‘  91 | ’  92 | “  93 | ”  94 | •  95 | –  96 | —  97 | 98 | ™  99 | љ  9A | ›  9B | њ  9C | ќ  9D | ћ  9E | џ  9F |
| **A** | NBSP  A0 | Ў  A1 | ў  A2 | Ј  A3 | ¤  A4 | Ґ  A5 | ¦  A6 | §  A7 | Ё  A8 | ©  A9 | Є  AA | «  AB | ¬  AC | SHY  AD | ®  AE | Ї  AF |
| **B** | °  B0 | ±  B1 | І  B2 | і  B3 | ґ  B4 | µ  B5 | ¶  B6 | ·  B7 | ё  B8 | №  B9 | є  BA | »  BB | ј  BC | Ѕ  BD | ѕ  BE | ї  BF |
| **C** | А  C0 | Б  C1 | В  C2 | Г  C3 | Д  C4 | Е  C5 | Ж  C6 | З  C7 | И  C8 | Й  C9 | К  CA | Л  CB | М  CC | Н  CD | О  CE | П  CF |
| **D** | Р  D0 | С  D1 | Т  D2 | У  D3 | Ф  D4 | Х  D5 | Ц  D6 | Ч  D7 | Ш  D8 | Щ  D9 | Ъ  DA | Ы  DB | Ь  DC | Э  DD | Ю  DE | Я  DF |
| **E** | а  E0 | б  E1 | в  E2 | г  E3 | д  E4 | е  E5 | ж  E6 | з  E7 | и  E8 | й  E9 | к  EA | л  EB | м  EC | н  ED | о  EE | п  EF |
| **F** | р  F0 | с  F1 | т  F2 | у  F3 | ф  F4 | х  F5 | ц  F6 | ч  F7 | ш  F8 | щ  F9 | ъ  FA | ы  FB | ь  FC | э  FD | ю  FE | я  FF |

**ПРИЛОЖЕНИЕ 2**

**Таблица 5. Unicode: блок Controls and Basic Latin**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **A** | **B** | **C** | **D** | **E** | **F** |
| **000** | NUL  0000 | SOH  0001 | STX  0002 | ETX  0003 | EOT  0004 | ENQ  0005 | ACK  0006 | BEL  0007 | BS  0008 | HT  0009 | LF  000A | VT  000B | FF  000C | CR  000D | SO  000E | SI  000F |
| **001** | DLE  0010 | DC1  0011 | DC2  0012 | DC3  0013 | DC4  0014 | NAK  0015 | SYN  0016 | ETB  0017 | CAN  0018 | EM  0019 | SUB  001A | ESC  001B | FS  001C | GS  001D | RS  001E | US  001F |
| **002** | SP  0020 | !  0021 | "  0022 | #  0023 | $  0024 | %  0025 | &  0026 | '  0027 | (  0028 | )  0029 | \*  002A | +  002B | ,  002C | -  002D | .  002E | /  002F |
| **003** | 0  0030 | 1  0031 | 2  0032 | 3  0033 | 4  0034 | 5  0035 | 6  0036 | 7  0037 | 8  0038 | 9  0039 | :  003A | ;  003B | <  003C | =  003D | >  003E | ?  003F |
| **004** | @  0040 | A  0041 | B  0042 | C  0043 | D  0044 | E  0045 | F  0046 | G  0047 | H  0048 | I  0049 | J  004A | K  004B | L  004C | M  004D | N  004E | O  004F |
| **005** | P  0050 | Q  0051 | R  0052 | S  0053 | T  0054 | U  0055 | V  0056 | W  0057 | X  0058 | Y  0059 | Z  005A | [  005B | \  005C | ]  005D | ^  005E | \_  005F |
| **006** | `  0060 | a  0061 | b  0062 | c  0063 | d  0064 | e  0065 | f  0066 | g  0067 | h  0068 | i  0069 | j  006A | k  006B | l  006C | m  006D | n  006E | o  006F |
| **007** | p  0070 | q  0071 | r  0072 | s  0073 | t  0074 | u  0075 | v  0076 | w  0077 | x  0078 | y  0079 | z  007A | {  007B | |  007C | }  007D | ~  007E | DEL  007F |

**Таблица 6. Unicode: блок Cyrillic**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **A** | **B** | **C** | **D** | **E** | **F** |
| **040** | Ѐ  0400 | Ё  0401 | Ђ  0402 | Ѓ  0403 | Є  0404 | Ѕ  0405 | І  0406 | Ї  0407 | Ј  0408 | Љ  0409 | Њ  040A | Ћ  040B | Ќ  040C | Ѝ  040D | Ў  040E | Џ  040F |
| **041** | А  0410 | Б  0411 | В  0412 | Г  0413 | Д  0414 | Е  0415 | Ж  0416 | З  0417 | И  0418 | Й  0419 | К  041A | Л  041B | М  041C | Н  041D | О  041E | П  041F |
| **042** | Р  0420 | С  0421 | Т  0422 | У  0423 | Ф  0424 | Х  0425 | Ц  0426 | Ч  0427 | Ш  0428 | Щ  0429 | Ъ  042A | Ы  042B | Ь  042C | Э  042D | Ю  042E | Я  042F |
| **043** | а  0430 | б  0431 | в  0432 | г  0433 | д  0434 | е  0435 | ж  0436 | з  0437 | и  0438 | й  0439 | к  043A | л  043B | м  043C | н  043D | о  043E | п  043F |
| **044** | р  0440 | с  0441 | т  0442 | у  0443 | ф  0444 | х  0445 | ц  0446 | ч  0447 | ш  0448 | щ  0449 | ъ  044A | ы  044B | ь  044C | э  044D | ю  044E | я  044F |
| **045** | ѐ  0450 | ё  0451 | ђ  0452 | ѓ  0453 | є  0454 | ѕ  0455 | і  0456 | ї  0457 | ј  0458 | љ  0459 | њ  045A | ћ  045B | ќ  045C | ѝ  045D | ў  045E | џ  045F |
| **046** | Ѡ  0460 | ѡ  0461 | Ѣ  0462 | ѣ  0463 | Ѥ  0464 | ѥ  0465 | Ѧ  0466 | ѧ  0467 | Ѩ  0468 | ѩ  0469 | Ѫ  046A | ѫ  046B | Ѭ  046C | ѭ  046D | Ѯ  046E | ѯ  046F |
| **047** | Ѱ  0470 | ѱ  0471 | Ѳ  0472 | ѳ  0473 | Ѵ  0474 | ѵ  0475 | Ѷ  0476 | ѷ  0477 | Ѹ  0478 | ѹ  0479 | Ѻ  047A | ѻ  047B | Ѽ  047C | ѽ  047D | Ѿ  047E | ѿ  047F |
| **048** | Ҁ  0480 | ҁ  0481 | ҂  0482 | ҃  0483 | ҄  0484 | ҅  0485 | ҆  0486 | ҇  0487 | ҈  0488 | ҉  0489 | Ҋ  048A | ҋ  048B | Ҍ  048C | ҍ  048D | Ҏ  048E | ҏ  048F |
| **049** | Ґ  0490 | ґ  0491 | Ғ  0492 | ғ  0493 | Ҕ  0494 | ҕ  0495 | Җ  0496 | җ  0497 | Ҙ  0498 | ҙ  0499 | Қ  049A | қ  049B | Ҝ  049C | ҝ  049D | Ҟ  049E | ҟ  049F |
| **04A** | Ҡ  04A0 | ҡ  04A1 | Ң  04A2 | ң  04A3 | Ҥ  04A4 | ҥ  04A5 | Ҧ  04A6 | ҧ  04A7 | Ҩ  04A8 | ҩ  04A9 | Ҫ  04AA | ҫ  04AB | Ҭ  04AC | ҭ  04AD | Ү  04AE | ү  04AF |
| **04B** | Ұ  04B0 | ұ  04B1 | Ҳ  04B2 | ҳ  04B3 | Ҵ  04B4 | ҵ  04B5 | Ҷ  04B6 | ҷ  04B7 | Ҹ  04B8 | ҹ  04B9 | Һ  04BA | һ  04BB | Ҽ  04BC | ҽ  04BD | Ҿ  04BE | ҿ  04BF |
| **04C** | Ӏ  04C0 | Ӂ  04C1 | ӂ  04C2 | Ӄ  04C3 | ӄ  04C4 | Ӆ  04C5 | ӆ  04C6 | Ӈ  04C7 | ӈ  04C8 | Ӊ  04C9 | ӊ  04CA | Ӌ  04CB | ӌ  04CC | Ӎ  04CD | ӎ  04CE | ӏ  04CF |
| **04D** | Ӑ  04D0 | ӑ  04D1 | Ӓ  04D2 | ӓ  04D3 | Ӕ  04D4 | ӕ  04D5 | Ӗ  04D6 | ӗ  04D7 | Ә  04D8 | ә  04D9 | Ӛ  04DA | ӛ  04DB | Ӝ  04DC | ӝ  04DD | Ӟ  04DE | ӟ  04DF |
| **04E** | Ӡ  04E0 | ӡ  04E1 | Ӣ  04E2 | ӣ  04E3 | Ӥ  04E4 | ӥ  04E5 | Ӧ  04E6 | ӧ  04E7 | Ө  04E8 | ө  04E9 | Ӫ  04EA | ӫ  04EB | Ӭ  04EC | ӭ  04ED | Ӯ  04EE | ӯ  04EF |
| **04F** | Ӱ  04F0 | ӱ  04F1 | Ӳ  04F2 | ӳ  04F3 | Ӵ  04F4 | ӵ  04F5 | Ӷ  04F6 | ӷ  04F7 | Ӹ  04F8 | ӹ  04F9 | Ӻ  04FA | ӻ  04FB | Ӽ  04FC | ӽ  04FD | Ӿ  04FE | ӿ  04FF |