VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Unicode/UTF-8

3 laboratory work

Lecturer: doc. dr. Pavel Stefanovič

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Unicode (1)

- Unicode – a standard, which defines almost all languages alphabets and additional character encoding in computers.
- It was designed to replace the previously used a variety of limited character encodings. The first version of the standard (1.0) show up in 1995. In 2005, have been announced the 4.1 version. Currently, Unicode is the dominant standard for adapting the computer programs in many languages.

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Unicode (2)

- In Unicode each position corresponds to only one specific character, but in some cases, for one symbol is given a few positions.
- First 256 positions are identical to ISO 8859-1 encoding characters that are easier replacement of the existing Western European language versions.
- The Unicode standard provide not only letters and symbols, but also additional codes which helps to describe the characteristics of the character, text direction, and other uses.

http://unicode-table.com/en/

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

ASCII table

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
10	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
20	32	! 33	" 34	# 35	$ 36	% 37	& 38	' 39	(40	) 41	* 42	+ 43	, 44	- 45	. 46	/ 47
30	0 48	1 49	2 50	3 51	4 52	5 53	6 54	7 55	8 56	9 57	: 58	; 59	< 60	= 61	> 62	? 63
40	@ 64	A 65	B 66	C 67	D 68	E 69	F 70	G 71	H 72	I 73	J 74	K 75	L 76	M 77	N 78	O 79
50	P 80	Q 81	R 82	S 83	T 84	U 85	V 86	W 87	X 88	Y 89	Z 90	[91	\ 92	] 93	^ 94	_ 95
60	` 96	a 97	b 98	c 99	d 100	e 101	f 102	g 103	h 104	i 105	j 106	k 107	l 108	m 109	n 110	o 111
70	p 112	q 113	r 114	s 115	t 116	u 117	v 118	w 119	x 120	y 121	z 122	{ 123	| 124	} 125	~ 126	127

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

UTF-8 (1)

- UTF-8 is a Unicode encoding type, in which, for one symbol is given from 1 byte to 4 bytes.
  - **1 byte** holds the English alphabet characters, numbers, punctuation marks. It is coded 128 characters, whose codes belong to ASCII encoding. In Unicode table it occupies the area from U+0000 to U+007F.
  - **2 bytes** occupied by extended Latin characters (letters and Lithuanian), Greek, Armenian, Coptic, Hebrew and Arabic alphabets and Cyrillic letters.
  - **3 bytes** occupied by the letters of other writing systems (Japanese, Chinese, and other Asian nations).

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

UTF-8 (2)

- **4 bytes** holding the characters are very rare.
- **5 bytes** and **6 bytes** characters are provided in the initial specification, but in November of 2003, RFC 3629 document limited the UTF-8 characters to **4 bytes**.

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

UTF-8 (3)

Number of bytes	Bits for code point	First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
1	7	U+0000	U+007F	0xxxxxxx			
2	11	U+0080	U+07FF	110xxxxx	10xxxxxx		
3	16	U+0800	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
4	21	U+10000	U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

https://r12a.github.io/apps/conversion/

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

INT → UNICODE → UTF-8 (1)

1) Let say we have the symbol, which INT (decimal number) is equal: **121**
2) $121_{10} \rightarrow X_{16}$
3) $0079_{16} \rightarrow X_2$
4) $0000\ 0000\ 0111\ 1001_2$

Number of bytes	Bits for code point	First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
1	7	U+0000	U+007F	0xxxxxxx			
2	11	U+0080	U+07FF	110xxxxx	10xxxxxx		
3	16	U+0800	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
4	21	U+10000	U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

UTF-8
79

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

INT → UNICODE → UTF-8 (2)

1) Let say we have the symbol, which INT (decimal number) is equal: **288**

2) $288_{10} \rightarrow X_{16}$

3) $0120_{16} \rightarrow X_2$

4) $0000\ 0001\ 0010\ 0000_2$

Number of bytes	Bits for code point	First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
1	7	U+0000	U+007F	0xxxxxxx			
2	11	U+0080	U+07FF	110xxxxx	10xxxxxx		
3	16	U+0800	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
4	21	U+10000	U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

UTF-8
C4 A0

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

INT → UNICODE → UTF-8 (3)

1) Let say we have the symbol, which INT (decimal number) is equal: **2336**

2) $2336_{10} \rightarrow X_{16}$

3) $0920_{16} \rightarrow X_2$

4) $0000\ 1001\ 0010\ 0000_2$

Number of bytes	Bits for code point	First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
1	7	U+0000	U+007F	0xxxxxxx			
2	11	U+0080	U+07FF	110xxxxx	10xxxxxx		
3	16	U+0800	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
4	21	U+10000	U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

UTF-8
E0 A4 A0

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Pseudocode with possible solution (1)

IF from 0000 to 0080

UTF-8 will be calculated by: 0079 ∧ 007F → 79

- As we can see, the logical AND operation between your char Unicode and 007F always gives the right UTF-8 answer. Check the table:

http://www.garykessler.net/library/byte_logic_table.html

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Pseudocode with possible solution (2)

IF from 0080 to 0800

UTF-8 will be separate to two bytes. It means, you will have to calculate two parts.

1 byte: 11000000 ∨ (0123 ∧ 11111000000) → 6

0123 → 0000 0001 0010 0011 ∧ 11111000000 → 6 → 0000 0100

11000000 ∨ 0000 0100 → 1100 0100 → C4

2 byte: 1000000 ∨ (0123 ∧ 111111)

0123 ∧ 111111 → 0000 0001 0010 0011 ∧ 111111 → 1010 0011 → A3

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Pseudocode of possible solution (3)

- Similar calculation will be done when we have 3 bytes.
- Try to come up with it.
- The bounds will be from 800 to 10000.

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Code page 437 (1)

- Code page 437 is the character set of the original IBM PC (personal computer), or DOS.

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Code page 437 (2)

Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
00C7	00FC	00E9	00E2	00E4	00E0	00E5	00E7	00EA	00EB	00E8	00EF	00EE	00EC	00C4	00C5
128	*129*	*130*	*131*	*132*	*133*	*134*	*135*	*136*	*137*	*138*	*139*	*140*	*141*	*142*	*143*
É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	¢	£	¥	₧	ƒ
00C9	00E6	00C6	00F4	00F6	00F2	00FB	00F9	00FF	00D6	00DC	00A2	00A3	00A5	20A7	0192
144	*145*	*146*	*147*	*148*	*149*	*150*	*151*	*152*	*153*	*154*	*155*	*156*	*157*	*158*	*159*
á	í	ó	ú	ñ	Ñ	ª	º	¿	⌐	¬	½	¼	¡	«	»
00E1	00ED	00F3	00FA	00F1	00D1	00AA	00BA	00BF	2310	00AC	00BD	00BC	00A1	00AB	00BB
160	*161*	*162*	*163*	*164*	*165*	*166*	*167*	*168*	*169*	*170*	*171*	*172*	*173*	*174*	*175*
░	▒	▓	│	┤	╡	╢	╖	╕	╣	║	╗	╝	╜	╛	┐
2591	2592	2593	2502	2524	2561	2562	2556	2555	2563	2551	2557	255D	255C	255B	2510
176	*177*	*178*	*179*	*180*	*181*	*182*	*183*	*184*	*185*	*186*	*187*	*188*	*189*	*190*	*191*
└	┴	┬	├	─	┼	╞	╟	╚	╔	╩	╦	╠	═	╬	╧
2514	2534	252C	251C	2500	253C	255E	255F	255A	2554	2569	2566	2560	2550	256C	2567
192	*193*	*194*	*195*	*196*	*197*	*198*	*199*	*200*	*201*	*202*	*203*	*204*	*205*	*206*	*207*
╨	╤	╥	╙	╘	╒	╓	╫	╪	┘	┌	█	▄	▌	▐	▀
2568	2564	2565	2559	2558	2552	2553	256B	256A	2518	250C	2588	2584	258C	2590	2580
208	*209*	*210*	*211*	*212*	*213*	*214*	*215*	*216*	*217*	*218*	*219*	*220*	*221*	*222*	*223*
α	ß	Γ	π	Σ	σ	µ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩
03B1	00DF	0393	03C0	03A3	03C3	00B5	03C4	03A6	0398	03A9	03B4	221E	03C6	03B5	2229
224	*225*	*226*	*227*	*228*	*229*	*230*	*231*	*232*	*233*	*234*	*235*	*236*	*237*	*238*	*239*
≡	±	≥	≤	⌠	⌡	÷	≈	°	∙	·	√	ⁿ	²	■	NBSP
2261	00B1	2265	2264	2320	2321	00F7	2248	00B0	2219	00B7	221A	207F	00B2	25A0	00A0
240	*241*	*242*	*243*	*244*	*245*	*246*	*247*	*248*	*249*	*250*	*251*	*252*	*253*	*254*	*255*

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Code page 437 (3)

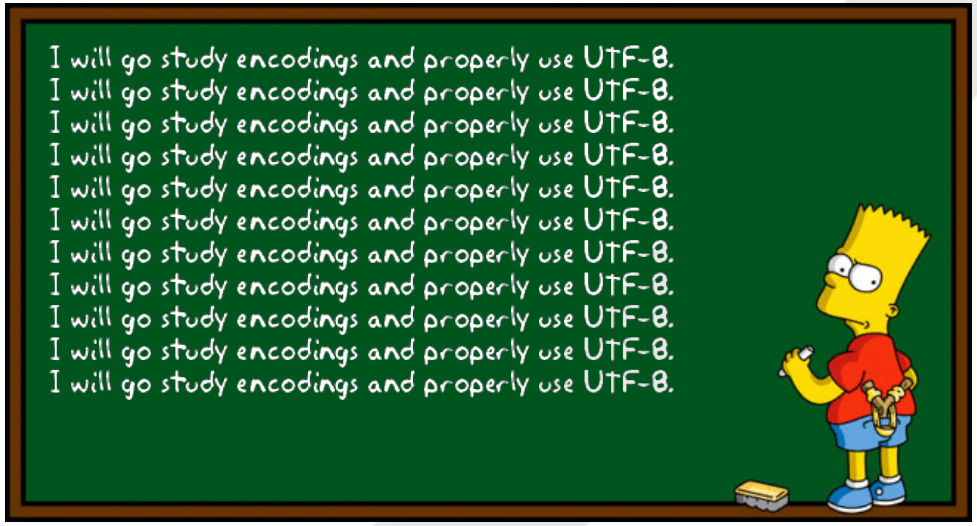
1. Read the file “386intel”.
2. Obtained the decimal expression of each symbol in the file.
3. Symbol are changed according to the CP437 table.
4. The new Unicode of char is assign.
5. The UTF-8 code is obtained and written to the new file.

P. S. if symbol ≤ 127, it just have to be rewritten to the file, no change.

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

3 laboratory work (1)

- **1 part (*0.3 point*):**
  - write a program, which allow to convert the decimal number to his Unicode and in the result show it in UTF-8 format;
  - show how it is working with a different symbols (use INT value).



VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

3 laboratory work (2)

- **2 part (*0.5 point*):**
  - convert file “386intel.txt” from the old 1 byte encoding system to Unicode by changing symbols to graphical equivalent. Use your created program from the first part.

You will need:

- the symbols table, where old code match the new one (it can be written in the program, but better is to use the separate file);
- program, which can read the file information by each byte and if the value of byte is ≥ 128, it have to change it to a new Unicode. Later, it have to be converter to UTF-8 and the results written to the new file or *stdout*.