Materiały do wykładu

2. Reprezentacja danych

Marcin Peczarski

Instytut Informatyki
Uniwersytet Warszawski

7 lutego 2012

Konwersje podstawy

- Między podstawą 2 a 16

$$(0010101001111110)_2 = (2A7E)_{16}$$

0010	1010	0111	1110
2	A	7	E

- Między podstawą 2 a 8

$$(0010101001111110)_2 = (025176)_8$$

0	010	101	001	111	110
0	2	5	1	7	6

Liczby całkowite (1)

$$b_{n-1}b_{n-2}\ldots b_2b_1b_0$$

naturalny kod binarny (NKB)	$\sum_{i=0}^{n-1} b_i 2^i$
moduł ze znakiem (MZ)	$(-1)^{b_{n-1}} \cdot \sum_{i=0}^{n-2} b_i 2^i$
uzupełnieniowy do dwójki (U2)	$-b_{n-1}2^{n-1} + \sum_{i=0}^{n-2} b_i 2^i$
uzupełnieniowy do jedynek (U1)	$\sum_{i=0}^{n-2} (b_i - b_{n-1}) 2^i$
spolaryzowany (ang. *biased*)	$-B + \sum_{i=0}^{n-1} b_i 2^i$

Liczby całkowite (2)

$b_{n-1}b_{n-2}\ldots b_2b_1b_0$ (n podzielne przez 4)

BCD (ang. *binary coded decimal*) $\sum_{j=0}^{n/4-1}\sum_{i=0}^{3} b_{i+4j}2^i10^j$

Liczby całkowite (3)

$b_7 \ldots b_1 b_0$	NKB	MZ	U2	U1	$B = 127$	BCD
00000000	0	0	0	0	-127	0
00000001	1	1	1	1	-126	1
⋮						
01111110	126	126	126	126	-1	–
01111111	127	127	127	127	0	–
10000000	128	0	-128	-127	1	80
10000001	129	-1	-127	-126	2	81
⋮						
11111110	254	-126	-2	-1	127	–
11111111	255	-127	-1	0	128	–

Negacja arytmetyczna

- MZ – negujemy bit znaku:

$$\overline{b_{n-1}}\, b_{n-2} \ldots b_2\, b_1\, b_0$$

- U2 – negujemy wszystkie bity i dodajemy jeden:

$$\overline{b_{n-1}}\, \overline{b_{n-2}} \ldots \overline{b_2}\, \overline{b_1}\, \overline{b_0} + 1$$

- U1 – negujemy wszystkie bity:

$$\overline{b_{n-1}}\, \overline{b_{n-2}} \ldots \overline{b_2}\, \overline{b_1}\, \overline{b_0}$$

- Spolaryzowany – wykonujemy odejmowanie (w NKB):

$$2B - b_{n-1}\, b_{n-2} \ldots b_2\, b_1\, b_0$$

Dlaczego używamy U2?

- Oznaczmy interpretację n-bitowej liczby $x_{n-1} \dots x_1 x_0$ w NKB i U2 odpowiednio przez X_{NKB} i X_{U2}.

- Mamy wtedy:

$$X_{\mathrm{NKB}} \equiv X_{\mathrm{U2}} \pmod{2^n},$$

$$X_{\mathrm{NKB}} + Y_{\mathrm{NKB}} \equiv X_{\mathrm{U2}} + Y_{\mathrm{U2}} \pmod{2^n},$$

$$X_{\mathrm{NKB}} - Y_{\mathrm{NKB}} \equiv X_{\mathrm{U2}} - Y_{\mathrm{U2}} \pmod{2^n},$$

$$X_{\mathrm{NKB}} \cdot Y_{\mathrm{NKB}} \equiv X_{\mathrm{U2}} \cdot Y_{\mathrm{U2}} \pmod{2^n}.$$

- Jeśli wynik działania mieści się w n bitach, to nie ma znaczenia, czy interpretujemy liczby w NKB, czy w U2.

- Zatem te same układy arytmetyczne mogą być użyte do dodawania, odejmowania i mnożenia w NKB i U2.

Mnożenie i dzielenie przez 2

- Mnożenie – przesuwamy o jeden bit w lewo, najmłodszy bit zerujemy:

$$b_{n-2}\, b_{n-3} \ldots b_1\, b_0\, 0$$

- Dzielenie w NKB – przesuwamy o jeden bit w prawo, najstaszy bit zerujemy:

$$0\, b_{n-1} \ldots b_3\, b_2\, b_1$$

- Dzielenie w U2 – przesuwamy o jeden bit w prawo, bit znaku powielamy:

$$b_{n-1}\, b_{n-1} \ldots b_3\, b_2\, b_1$$

Przeniesienie w NKB i nadmiar w U2

- Jeśli wynik dodawania lub odejmowania dwóch n-bitowych liczb w NKB nie może być zapisany na n-bitach (jest za duży lub ujemny), to ustawiany jest bit przeniesienia (zwykle oznaczany C).

- Jeśli wynik dodawania lub odejmowania dwóch n-bitowych liczb w U2 nie może być zapisany na n-bitach, to ustawiany jest bit nadmiaru (zwykle oznaczany V).

Przeniesienie i nadmiar

odejmowanie	NKB	U2
10000010	130	−126
10000001	129	−127
00000001	1	1

$C = 0,\ V = 0$

odejmowanie	NKB	U2
10000010	130	−126
00000011	3	3
01111111	127	127

$C = 0,\ V = 1$

odejmowanie	NKB	U2
10000001	129	−127
10000010	130	−126
11111111	255	−1

$C = 1,\ V = 0$

odejmowanie	NKB	U2
00000011	3	3
10000010	130	−126
10000001	129	−127

$C = 1,\ V = 1$

Porównywanie liczb w NKB i U2

- Wykonujemy odejmowanie $X - Y$, ale nie zapisujemy wyniku.
- Zapisujemy:
  - czy wynik jest zerowy (bit Z),
  - czy wystąpiło przeniesienie (bit C),
  - czy wynik jest ujemny – najstarszy bit wyniku (bit S),
  - czy wystąpił nadmiar (bit V).
- $X = Y$ wtw, gdy $Z = 1$.
- $X_{\mathrm{NKB}} < Y_{\mathrm{NKB}}$ wtw, gdy $C = 1$.
- $X_{\mathrm{U2}} < Y_{\mathrm{U2}}$ wtw, gdy $S \oplus V = 1$.
- Pozostałe relacje ($\neq$, $>$, $\geqslant$, $\leqslant$) można łatwo uzyskać z powyższych.

Liczby zmiennopozycyjne IEEE-754

pojedyncza precyzja	31	30 ... 23	22 ...	0	$B = 127$
podwójna precyzja	63	62 ... 52	51 ...	0	$B = 1023$

pola wykładnika i mantysy	rodzaj wartości	wartość
$W \neq 0\ldots0,\ W \neq 1\ldots1$	znormalizowane	$(-1)^S \cdot (1,M)_2 \cdot 2^{W-B}$
$W = 0\ldots0,\ M \neq 0\ldots0$	zdenormalizowane	$(-1)^S \cdot (0,M)_2 \cdot 2^{1-B}$
$W = 0\ldots0,\ M = 0\ldots0$	zera	0
$W = 1\ldots1,\ M = 0\ldots0$	nieskończoności	$(-1)^S \cdot \infty$
$W = 1\ldots1,\ M \neq 0\ldots0$	nieliczby	nan, NaN

Porównywanie liczb zmiennopozycyjnych

- Przyjęta reprezentacja umożliwia prostą metodę porównywania – wystarczy porównywać kolejne bity od lewej do prawej.
- Nieskończoność jest większa od dowolnej liczby rzeczywistej.
- Minus nieskończoność jest mniejsza od dowolnej liczby rzeczywistej.
- Wynik porównywania dwóch wartości w formacie IEEE-754, oprócz $<$, $=$, $>$, może też być *nieuporządkowane*, gdy co najmniej jedna z porównywanych wartości jest nieliczbą.

Zaokrąglanie

- Do najbliższej lub parzystej (ang. *round to nearest or even*) – najlepszy sposób zaokrąglania
- Ucięcie (ang. *chop*, *truncate*) do liczby bliższej zeru – stosowany często w językach wysokiego poziomu
- W dół (ang. *round down*)
- W górę (ang. *round up*)

Zaokrąglanie w C

▷ Program

```
#include "stdio.h"

int main() {
  double x = 0.478;
  double a = 1000.0;
  int ix;

  ix = (int) a * x;
  printf("x = %f, (int) 1000.0 * x = %d\n", x, ix);
  return 0;
}
```

▷ wypisuje

```
x = 0.478000, (int) 1000.0 * x = 477
```

Napisy (1)

2.15

437 500 500V1 850 851 852 855 856 857 860 861 862 863 864 865 866 866NAV 869 874 904 1026 1046 1047 8859_1 8859_2 8859_3 8859_4 8859_5 8859_6 8859_7 8859_8 8859_9 10646-1:1993 10646-1:1993/UCS4/ ANSI_X3.4-1968 ANSI_X3.4-1986 ANSI_X3.4 ANSI_X3.110-1983 ANSI_X3.110 ARABIC ARABIC7 ARMSCII-8 ASCII ASMO-708 ASMO_449 BALTIC BIG-5 BIG-FIVE BIG5-HKSCS BIG5 BIG5HKSCS BIGFIVE BS_4730 CA CN-BIG5 CN-GB CN CP-AR CP-GR CP-HU CP037 CP038 CP273 CP274 CP275 CP278 CP280 CP281 CP282 CP284 CP285 CP290 CP297 CP367 CP420 CP423 CP424 CP437 CP500 CP737 CP775 CP813 CP819 CP850 CP851 CP852 CP855 CP856 CP857 CP860 CP861 CP862 CP863 CP864 CP865 CP866 CP866NAV CP868 CP869 CP870 CP871 CP874 CP875 CP880 CP891 CP903 CP904 CP905 CP912 CP915 CP916 CP918 CP920 CP922 CP930 CP932 CP933 CP935 CP936 CP937 CP939 CP949 CP950 CP1004 CP1025 CP1026 CP1046 CP1047 CP1070 CP1079 CP1081 CP1084 CP1089 CP1122 CP1124 CP1125 CP1129 CP1132 CP1133 CP1137 CP1153 CP1154 CP1155 CP1156 CP1157 CP1158 CP1160 CP1161 CP1162 CP1163 CP1164 CP1250 CP1251 CP1252 CP1253 CP1254 CP1255 CP1256 CP1257 CP1258 CP1361 CP10007 CPIBM861 CSA7-1 CSA7-2 CSASCII CSA_T500-1983 CSA_T500 CSA_Z243.4-1985-1 CSA_Z243.4-1985-2 CSA_Z243.419851 CSA_Z243.419852 CSDECMCS CSEBCDICATDE CSEBCDICATDEA CSEBCDICCAFR CSEBCDICDKNO CSEBCDICDKNOA CSEBCDICES CSEBCDICESA CSEBCDICESS CSEBCDICFISE CSEBCDICFISEA CSEBCDICFR CSEBCDICIT CSEBCDICPT CSEBCDICUK CSEBCDICUS CSEUCKR CSEUCPKDFMTJAPANESE CSGB2312 CSHPROMAN8 CSIBM037 CSIBM038 CSIBM273 CSIBM274 CSIBM275 CSIBM277 CSIBM278 CSIBM280 CSIBM281 CSIBM284 CSIBM285 CSIBM290 CSIBM297 CSIBM420 CSIBM423 CSIBM424 CSIBM500 CSIBM851 CSIBM855 CSIBM856 CSIBM857 CSIBM860 CSIBM863 CSIBM864 CSIBM865 CSIBM866 CSIBM868 CSIBM869 CSIBM870 CSIBM871 CSIBM880 CSIBM891 CSIBM903 CSIBM904 CSIBM905 CSIBM918 CSIBM922 CSIBM930 CSIBM932 CSIBM933 CSIBM935 CSIBM937 CSIBM939 CSIBM943 CSIBM1025 CSIBM1026 CSIBM1122 CSIBM1124 CSIBM1129 CSIBM1132 CSIBM1133 CSIBM1137 CSIBM1153 CSIBM1154 CSIBM1155 CSIBM1156 CSIBM1157 CSIBM1158 CSIBM1160 CSIBM1161 CSIBM1163 CSIBM1164 CSIBM11621162 CSISO4UNITEDKINGDOM CSISO10SWEDISH CSISO11SWEDISHFORNAMES CSISO14JISC6220RO CSISO15ITALIAN CSISO16PORTUGESE CSISO17SPANISH CSISO18GREEK7OLD CSISO19LATINGREEK CSISO21GERMAN CSISO25FRENCH CSISO27LATINGREEK1 CSISO49INIS CSISO50INIS8 CSISO51INISCYRILLIC CSISO58GB1988 CSISO60DANISHNORWEGIAN CSISO60NORWEGIAN1 CSISO61NORWEGIAN2 CSISO69FRENCH CSISO84PORTUGUESE2 CSISO85SPANISH2 CSISO86HUNGARIAN CSISO88GREEK7 CSISO89ASMO449 CSISO90 CSISO92JISC62991984B CSISO99NAPLPS CSISO103T618BIT CSISO111ECMACYRILLIC CSISO121CANADIAN1 CSISO122CANADIAN2 CSISO139CSN369103 CSISO141JUSIB1002 CSISO143IECP271 CSISO150 CSISO150GREEKCCITT CSISO151CUBA CSISO153GOST1976874 CSISO646DANISH CSISO2022CN CSISO2022JP CSISO2022JP2 CSISO2022KR CSISO2033 CSISO5427CYRILLIC CSISO5427CYRILLIC1981 CSISO5428GREEK CSISO10367BOX CSISOLATIN1 CSISOLATIN2 CSISOLATIN3 CSISOLATIN4 CSISOLATIN5 CSISOLATINARABIC CSISOLATINCYRILLIC CSISOLATINGREEK CSISOLATINHEBREW CSKOI8R CSKSC5636 CSMACINTOSH CSNATSDANO CSNATSSEFI CSN_369103 CSPC8CODEPAGE437 CSPC775BALTIC CSPC850MULTILINGUAL CSPC862LATINHEBREW CSPCP852 CSSHIFTJIS CSUCS4 CSUNICODE CSWINDOWS31J CUBA CWI-2 CWI CYRILLIC DE DEC-MCS DEC DECMCS DIN_66003 DK DS2089 DS_2089 E13B/ EBCDIC-AT-DE-A EBCDIC-AT-DE EBCDIC-BE EBCDIC-BR EBCDIC-CA-FR EBCDIC-CP-AR1 EBCDIC-CP-AR2 EBCDIC-CP-BE EBCDIC-CP-CA EBCDIC-CP-CH EBCDIC-CP-DK EBCDIC-CP-ES EBCDIC-CP-FI EBCDIC-CP-FR EBCDIC-CP-GB EBCDIC-CP-GR EBCDIC-CP-HE EBCDIC-CP-IS EBCDIC-CP-IT EBCDIC-CP-NL EBCDIC-CP-NO EBCDIC-CP-ROECE EBCDIC-CP-SE EBCDIC-CP-TR EBCDIC-CP-US EBCDIC-CP-WT EBCDIC-CP-YU EBCDIC-CYRILLIC EBCDIC-DK-NO-A EBCDIC-DK-NO EBCDIC-ES-A EBCDIC-ES-S EBCDIC-ES EBCDIC-FI-SE-A EBCDIC-FI-SE EBCDIC-FR EBCDIC-GREEK EBCDIC-INT EBCDIC-INT1 EBCDIC-IS-FRISS EBCDIC-IT EBCDIC-JP-E EBCDIC-JP-KANA EBCDIC-PT EBCDIC-UK EBCDIC-US EBCDICATDE EBCDICATDEA EBCDICCAFR EBCDICDKNO EBCDICDKNOA EBCDICES EBCDICESA EBCDICESS EBCDICFISE EBCDICFISEA EBCDICFR EBCDICISFRISS EBCDICIT EBCDICPT EBCDICUK EBCDICUS ECMA-114 ECMA-118 ECMA-128 ECMA-CYRILLIC ECMACYRILLIC ELOT_928 ES ES2 EUC-CN EUC-JISX0213 EUC-JP-MS EUC-JP EUC-KR EUC-TW EUCCN EUCJP-MS EUCJP-OPEN EUCJP-WIN EUCJP EUCKR EUCTW FI FR GB GB2312 GB13000 GB18030 GBK GB_1988-80 GB_198880 GEORGIAN-ACADEMY GEORGIAN-PS GOST_19768-74 GOST_19768 GOST_1976874 GREEK-CCITT GREEK GREEK7-OLD GREEK7 GREEK7OLD GREEK8 GREEKCCITT HEBREW HP-ROMAN8 HPROMAN8 HU IBM-856 IBM-922 IBM-930 IBM-932 IBM-933 IBM-935 IBM-937 IBM-939 IBM-943 IBM-1025 IBM-1046 IBM-1047 IBM-1122 IBM-1124 IBM-1129 IBM-1132 IBM-1133 IBM-1137 IBM-1153 IBM-1154 IBM-1155 IBM-1156 IBM-1157 IBM-1158 IBM-1160 IBM-1161 IBM-1162 IBM-1163 IBM-1164 IBM037 IBM038 IBM256 IBM273 IBM274 IBM275 IBM277 IBM278 IBM280 IBM281 IBM284 IBM285 IBM290 IBM297 IBM367 IBM420 IBM423 IBM424 IBM437 IBM500 IBM775 IBM813 IBM819 IBM848 IBM850 IBM851 IBM852 IBM855 IBM856 IBM857 IBM860 IBM861 IBM862 IBM863 IBM864 IBM865 IBM866 IBM866NAV IBM868 IBM869 IBM870 IBM871 IBM874 IBM875 IBM880 IBM891 IBM903 IBM904 IBM905 IBM912 IBM915 IBM916 IBM918 IBM920 IBM922 IBM930 IBM932 IBM933 IBM935 IBM937 IBM939 IBM943 IBM1004 IBM1025 IBM1026 IBM1046 IBM1047 IBM1089 IBM1122 IBM1124 IBM1129 IBM1132 IBM1133 IBM1137 IBM1153 IBM1154 IBM1155 IBM1156 IBM1157 IBM1158 IBM1160 IBM1161 IBM1162 IBM1163 IBM1164 IEC_P27-1 IEC_P271 INIS-8 INIS-CYRILLIC INIS INIS8 INISCYRILLIC ISIRI-3342 ISIRI3342 ISO-2022-CN-EXT ISO-2022-CN ISO-2022-JP-2 ISO-2022-JP-3 ISO-2022-JP ISO-2022-KR ISO-8859-1 ISO-8859-2 ISO-8859-3 ISO-8859-4 ISO-8859-5 ISO-8859-6 ISO-8859-7 ISO-8859-8 ISO-8859-9 ISO-8859-10 ISO-8859-11 ISO-8859-13 ISO-8859-14 ISO-8859-15 ISO-8859-16 ISO-10646 ISO-10646/UCS2/ ISO-10646/UCS4/ ISO-10646/UTF-8/ ISO-10646/UTF8/ ISO-CELTIC ISO-IR-4 ISO-IR-6 ISO-IR-8-1 ISO-IR-9-1 ISO-IR-10 ISO-IR-11 ISO-IR-14 ISO-IR-15 ISO-IR-16 ISO-IR-17 ISO-IR-18 ISO-IR-19 ISO-IR-21 ISO-IR-25 ISO-IR-27 ISO-IR-37 ISO-IR-49 ISO-IR-50 ISO-IR-51 ISO-IR-54 ISO-IR-55 ISO-IR-57 ISO-IR-60 ISO-IR-61 ISO-IR-69 ISO-IR-84 ISO-IR-85 ISO-IR-86 ISO-IR-88 ISO-IR-89 ISO-IR-90 ISO-IR-92 ISO-IR-98 ISO-IR-99 ISO-IR-100 ISO-IR-101 ISO-IR-103 ISO-IR-109 ISO-IR-110 ISO-IR-111 ISO-IR-121 ISO-IR-122 ISO-IR-126 ISO-IR-127 ISO-IR-138 ISO-IR-139 ISO-IR-141 ISO-IR-143 ISO-IR-144 ISO-IR-148 ISO-IR-150 ISO-IR-151 ISO-IR-153 ISO-IR-155 ISO-IR-156 ISO-IR-157 ISO-IR-166 ISO-IR-179 ISO-IR-193 ISO-IR-197 ISO-IR-199 ISO-IR-203 ISO-IR-209 ISO-IR-226 ISO646-CA ISO646-CA2 ISO646-CN ISO646-CU ISO646-DE ISO646-DK ISO646-ES ISO646-ES2 ISO646-FI ISO646-FR ISO646-FR1 ISO646-GB ISO646-HU ISO646-IT ISO646-JP-OCR-B ISO646-JP ISO646-KR ISO646-NO ISO646-NO2 ISO646-PT ISO646-PT2 ISO646-SE ISO646-SE2 ISO646-US ISO646-YU ISO2022CN ISO2022CNEXT ISO2022JP ISO2022JP2 ISO2022KR ISO6937 ISO8859-1 ISO8859-2 ISO8859-3 ISO8859-4 ISO8859-5 ISO8859-6 ISO8859-7 ISO8859-8 ISO8859-9 ISO8859-10 ISO8859-11 ISO8859-13 ISO8859-14 ISO8859-15 ISO8859-16 ISO88591 ISO88592 ISO88593 ISO88594 ISO88595 ISO88596 ISO88597 ISO88598 ISO88599 ISO885910 ISO885911 ISO885913 ISO885914 ISO885915 ISO885916 ISO_646.IRV:1991 ISO_2033-1983 ISO_2033 ISO_5427-EXT ISO_5427 ISO_5427:1981 ISO_5427EXT ISO_5428 ISO_5428:1980 ISO_6937-2 ISO_6937-2:1983 ISO_6937 ISO_6937:1992 ISO_8859-1 ISO_8859-1:1987 ISO_8859-2 ISO_8859-2:1987 ISO_8859-3 ISO_8859-3:1988 ISO_8859-4 ISO_8859-4:1988 ISO_8859-5 ISO_8859-5:1988 ISO_8859-6 ISO_8859-6:1987 ISO_8859-7 ISO_8859-7:1987 ISO_8859-7:2003 ISO_8859-8 ISO_8859-8:1988 ISO_8859-9 ISO_8859-9:1989 ISO_8859-10 ISO_8859-10:1992 ISO_8859-14 ISO_8859-14:1998 ISO_8859-15 ISO_8859-15:1998 ISO_8859-16 ISO_8859-16:2001 ISO_9036 ISO_10367-BOX ISO_10367BOX ISO_69372 IT JIS_C6220-1969-RO JIS_C6229-1984-B JIS_C62201969RO JIS_C62291984B JOHAB JP-OCR-B JP JS JUS_I.B1.002 KOI-7 KOI-8 KOI8-R KOI8-T KOI8-U KOI8 KOI8R KOI8U KSC5636 L1 L2 L3 L4 L5 L6 L7 L8 L10 LATIN-9 LATIN-GREEK-1 LATIN-GREEK LATIN1 LATIN2 LATIN3 LATIN4 LATIN5 LATIN6 LATIN7 LATIN8 LATIN10 LATINGREEK LATINGREEK1 MAC-CYRILLIC MAC-IS MAC-SAMI MAC-UK MAC MACCYRILLIC MACINTOSH MACIS MACUK MACUKRAINIAN MS-ANSI MS-ARAB MS-CYRL MS-EE MS-GREEK MS-HEBR MS-MAC-CYRILLIC MS-TURK MS932 MS936 MSCP949 MSCP1361 MSMACCYRILLIC MSZ_7795.3 MS_KANJI NAPLPS NATS-DANO NATS-SEFI NATSDANO NATSSEFI NC_NC0010 NC_NC00-10 NC_NC00-10:81 NF_Z_62-010 NF_Z_62-010_(1973) NF_Z_62-010_1973 NF_Z_62010 NF_Z_62010_1973 NO NO2 NS_4551-1 NS_4551-2 NS_45511 NS_45512 OS2LATIN1 OSF00010001 OSF00010002 OSF00010003 OSF00010004 OSF00010005 OSF00010006 OSF00010007 OSF00010008 OSF00010009 OSF0001000A OSF00010020 OSF00010100 OSF00010101 OSF00010102 OSF00010104 OSF00010105 OSF00010106 OSF00030010 OSF0004000A OSF0005000A OSF05010001 OSF100201A4 OSF100201A8 OSF100201B5 OSF100201F4 OSF100203B5 OSF1002011C OSF1002011D OSF1002035D OSF1002035E OSF1002035F OSF1002036B OSF1002037B OSF10010001 OSF10020025 OSF10020111 OSF10020115 OSF10020116 OSF10020118 OSF10020122 OSF10020129 OSF10020352 OSF10020354 OSF10020357 OSF10020359 OSF10020360 OSF10020364 OSF10020365 OSF10020366 OSF10020367 OSF10020370 OSF10020387 OSF10020388 OSF10020396 OSF10020402 OSF10020417 PT PT2 PT154 R8 RK1048 ROMAN8 RUSCII SE SE2 SEN_850200_B SEN_850200_C SHIFT-JIS SHIFT_JIS SHIFT_JISX0213 SJIS-OPEN SJIS-WIN SJIS SS636127 STRK1048-2002 ST_SEV_358-88 T.61-8BIT T.61 T.618BIT TCVN-5712 TCVN TCVN5712-1 TCVN5712-1:1993 TIS-620 TIS620-0 TIS620.2529-1 TIS620.2533-0 TIS620 TS-5881 TSCII UCS-2 UCS-2BE UCS-2LE UCS-4 UCS-4BE UCS-4LE UCS2 UCS4 UHC UJIS UK UNICODE UNICODEBIG UNICODELITTLE US-ASCII US UTF-7 UTF-8 UTF-16 UTF-16BE UTF-16LE UTF-32 UTF-32BE UTF-32LE UTF7 UTF8 UTF16 UTF16BE UTF16LE UTF32 UTF32BE UTF32LE VISCII WCHAR_T WIN-SAMI-2 WINBALTRIM WINDOWS-31J WINDOWS-874 WINDOWS-936 WINDOWS-1250 WINDOWS-1251 WINDOWS-1252 WINDOWS-1253 WINDOWS-1254 WINDOWS-1255 WINDOWS-1256 WINDOWS-1257 WINDOWS-1258 WINSAMI2 WS2 YU

Napisy (2)

ASCII – American Standard Code for Information Interchange

NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
␣	!	"	#	$	%	&	’	(	)	*	+	,	-	.	/
0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
‘	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

- ISO 8859-1, ISO 8859-2, ...
- cp1250, cp1251, ...
- ...

EBCDIC – Extended Binary Coded Decimal Interchange Code

NUL	SOH	STX	ETX	SEL	HT	RNL	DEL	GE	SPS	RPT	VT	FF	CR	SO	SI
DLE	DC1	DC2	DC3	RES	NL	BS	POC	CAN	EM	UBS	CU1	IFS	IGS	IRS	IUS
DS	SOS	FS	WUS	BYP	LF	ETB	ESC	SA	SFE	SM	CSP	MFA	ENQ	ACK	BEL
		SYN	IR	PP	TRN	NBS	EOT	SBS	IT	RFF	CU3	DC4	NAK		SUB
SP	RSP									[	.	<	(	+	!
&									ß	]	$	*	)	;	^
-	/										,	%	_	>	?
ø									‘	:	#	@	’	=	"
Ø	a	b	c	d	e	f	g	h	i	«	»				±
°	j	k	l	m	n	o	p	q	r	ª	º	æ	¸	Æ	
μ	~	s	t	u	v	w	x	y	z	¡	¿				
			·	©	§	¶	¼	½	¾	¬	|	-	¨	´	×
{	A	B	C	D	E	F	G	H	I	SHY					
}	J	K	L	M	N	O	P	Q	R	¹					
\	÷	S	T	U	V	W	X	Y	Z	²					
0	1	2	3	4	5	6	7	8	9	³					EO

UTF-8 – 8-bit Unicode Transformation Format

▷ ISO/IEC 10646 Universal Multiple-Octet Coded Character Set

Ą 104 ą 105	Ć 106 ć 107	Ę 118 ę 119	Ł 141 ł 142	Ń 143 ń 144	Ó D3 ó F3	Ś 15A ś 15B	Ź 179 ź 17A	Ż 17B ż 17C

▷ RFC 3629

zakres znaków	l. bitów	sekwencja oktetów UTF-8
0–7F	7	0xxxxxxx
80–7FF	11	110xxxxx 10xxxxxx
800–FFFF	16	1110xxxx 10xxxxxx 10xxxxxx
10000–10FFFF	21	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

Porządek bajtów

$\texttt{ABCD0123}_{16}$

cienkokońcówkowe
(ang. *little-endian*)

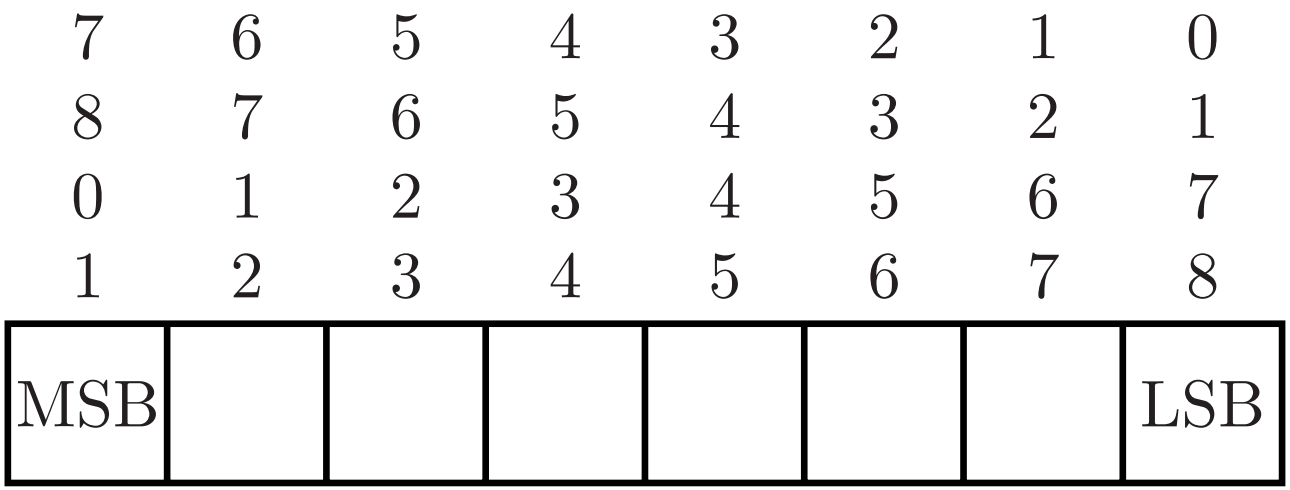
adres + 3	AB
adres + 2	CD
adres + 1	01
adres	23

grubokońcówkowe
(ang. *big-endian*)

adres + 3	23
adres + 2	01
adres + 1	CD
adres	AB

dwukońcówkowe (ang. *bi-endian*)

Porządek bitów



MSB – most significant bit
LSB – least significant bit

Typowe rozmiary typów prostych

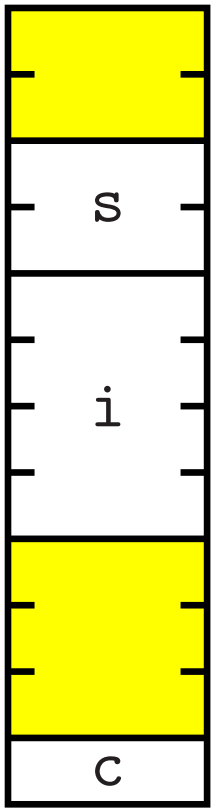
typ	maszyna			
	8-bitowa	16-bitowa	32-bitowa	64-bitowa
`char`	1	1	1	1
`short`	2	2	2	2
`int`	2	2	4	4
`long`	4	4	4	8
`void*`	2	2 lub 4	4	8
`float`	4	4	4	4
`double`	–	8	8	8

Wyrównywanie danych

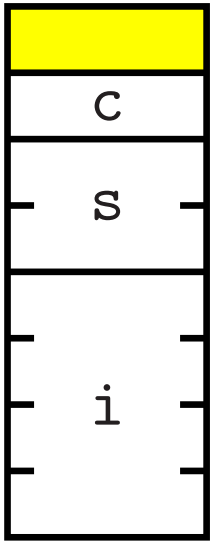
- Ma na celu optymalizację dostępów do pamięci.
- Adres, pod którym w pamięci umieszczana jest zmienna typu prostego, jest wielokrotnością rozmiaru:
  - słowa maszynowego,
  - typu.
- Tablica elementów typu prostego wyrównywana jest w całości wg rozmiaru typu składowego – nie zawiera dziur.

Wyrównywanie danych – typy złożone


```
typedef struct {
  char  c;
  int   i;
  short s;
} s1;
```

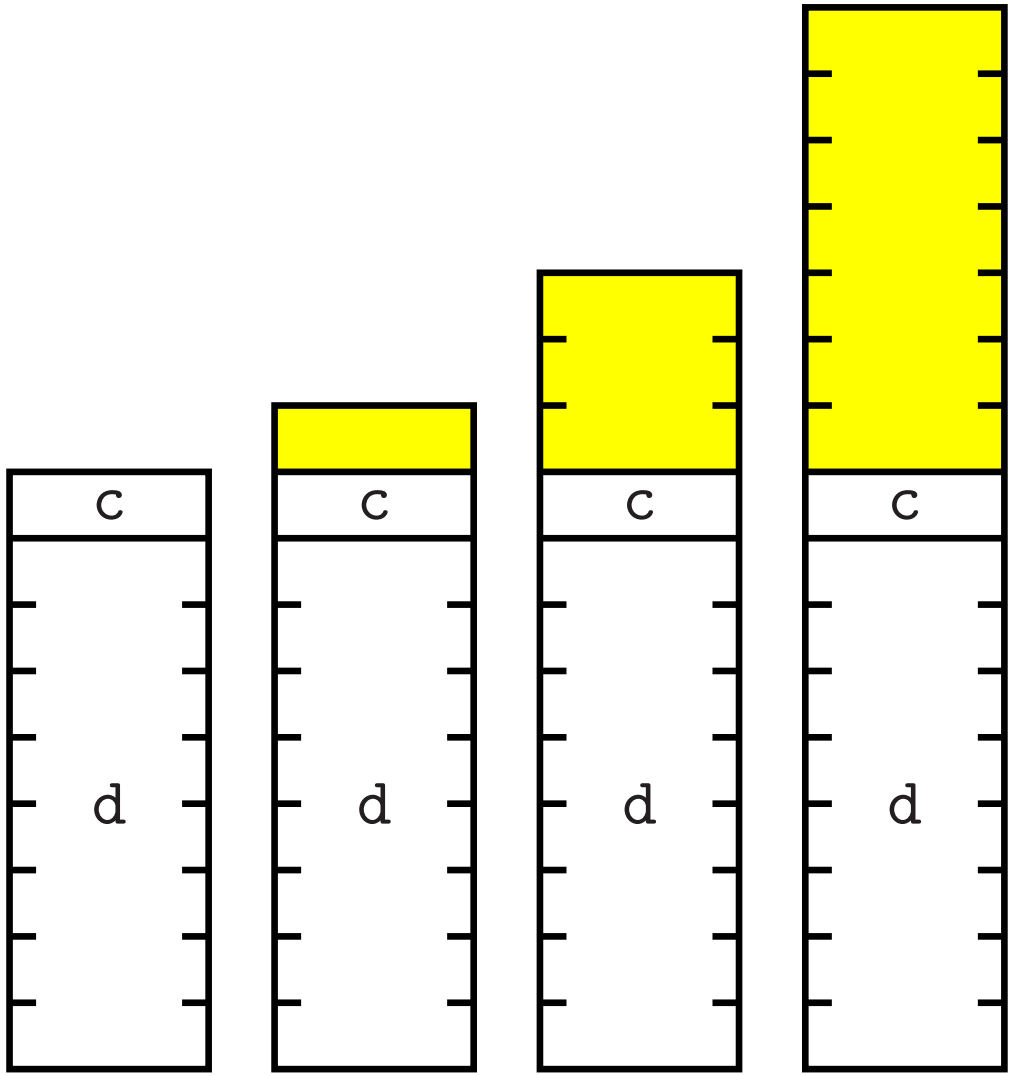



```
typedef struct {
  int   i;
  short s;
  char  c;
} s2;
```



Wyrównywanie danych zależne od kompilatora


```
typedef struct {
  double d;
  char   c;
} s3;
```



Przedrostki

kilo	k	10^3	(2^{10})			
mega	M	10^6	$(2^{20},$	$10^3 \cdot 2^{10})$		
giga	G	10^9	$(2^{30},$	$10^6 \cdot 2^{10},$	$10^3 \cdot 2^{20})$	
tera	T	10^{12}	$(2^{40},$	$10^9 \cdot 2^{10},$	$10^6 \cdot 2^{20},$	$10^3 \cdot 2^{30})$
peta	P	10^{15}	$(2^{50},$	$\ldots)$		
kibi	Ki	2^{10}				
mebi	Mi	2^{20}				
gibi	Gi	2^{30}				
tebi	Ti	2^{40}				
pebi	Pi	2^{50}				
eksbi	Ei	2^{60}				
zebi	Zi	2^{70}				
jobi	Yi	2^{80}				