

Header limits.h

Header	Opis zawartych funkcji	Przykładowe funkcje
limits.h	Podaje ograniczenia zmiennych	CHAR_BIT, SCHAR_MIN, SCHAR_MAX, UCHAR_MAX, CHAR_MIN, CHAR_MAX, SHRT_MIN, SHRT_MAX, USHRT_MAX, INT_MIN, INT_MAX, UINT_MAX, LONG_MIN, LONG_MAX, ULONG_MAX, LLONG_MIN, LLONG_MAX, ULLONG_MAX, MB_LEN_MAX

#include<limits.h>

- CHAR_BIT – Liczba bitów w znaku
- SCHAR_MIN – Minimalna wartość dla *signed char*
- SCHAR_MAX – Maksymalna wartość dla *signed char*
- UCHAR_MAX – Maksymalna wartość dla *unsigned char*
- CHAR_MIN – Minimalna wartość dla *char*
- CHAR_MAX – Maksymalna wartość dla *char*
- SHRT_MIN – Minimalna wartość dla *short int*
- SHRT_MAX – Maksymalna wartość dla *short int*
- USHRT_MAX – Maksymalna wartość dla *unsigned short int*
- INT_MIN – Minimalna wartość dla *int*
- INT_MAX – Maksymalna wartość dla *int*
- UINT_MAX – Maksymalna wartość dla *unsigned int*
- LONG_MIN – Minimalna wartość dla *long int*
- LONG_MAX – Maksymalna wartość dla *long int*
- ULONG_MAX – Maksymalna wartość dla *unsigned long int*
- LLONG_MIN – Minimalna wartość dla *long long int*
- LLONG_MAX – Maksymalna wartość dla *long long int*
- ULLONG_MAX – Maksymalna wartość dla *unsigned long long int*
- MB_LEN_MAX – Maksymalna wielobajtowa długość znaku we wszystkich ustawieniach regionalnych

=====

```
// Ograniczenia zakresu zmiennych

#include <stdio.h>
#include <limits.h>           // ten temat
#include <locale.h>           // dla 'setlocale()'

int main()
{ setlocale(LC_CTYPE, "Polish"); // polskie znaki

printf("\n - - - - - Ograniczenia typów - - - - - \n\n");
printf(" Liczba bitów w znaku = %d\n", CHAR_BIT);
printf(" Minimalna wartość dla SIGNED CHAR = %d\n", SCHAR_MIN);
printf(" Maksymalna wartość dla SIGNED CHAR = %d\n", SCHAR_MAX);
printf(" Maksymalna wartość dla UNSIGNED CHAR = %d\n", UCHAR_MAX);
printf(" Minimalna wartość dla CHAR = %d\n", CHAR_MIN);
printf(" Maksymalna wartość dla CHAR = %d\n", CHAR_MAX);
printf(" Minimalna wartość dla SHORT INT = %d\n", SHRT_MIN);
printf(" Maksymalna wartość dla SHORT INT = %d\n", SHRT_MAX);
```

```

printf(" Maksymalna wartość dla UNSIGNED SHORT INT = %u\n", USHRT_MAX);
printf(" Minimalna wartość dla INT = %d\n", INT_MIN);
printf(" Maksymalna wartość dla INT = %d\n", INT_MAX);
printf(" Maksymalna wartość dla UNSIGNED INT = %lu\n", UINT_MAX);
printf(" Minimalna wartość dla LONG INT = %ld\n", LONG_MIN);
printf(" Maksymalna wartość dla LONG INT = %ld\n", LONG_MAX);
printf(" Maksymalna wartość dla UNSIGNED LONG INT = %llu\n", ULONG_MAX);
printf(" Minimalna wartość dla LONG LONG INT = %lld\n", LLONG_MIN);
printf(" Maksymalna wartość dla LONG LONG INT = %lld\n", LLONG_MAX);
printf(" Maksymalna wartość dla UNSIGNED LONG LONG INT = %llu\n\n", ULLONG_MAX);
printf(" Maksymalna wielobajtowa długość znaku we wszystkich ustawieniach
regionalnych = %d\n", MB_LEN_MAX);

return 0;
}

```

Wynik działania programu:

```

- - - - - Ograniczenia typów - - - - -

Liczba bitów w znaku = 8
Minimalna wartość dla SIGNED CHAR = -128
Maksymalna wartość dla SIGNED CHAR = 127
Maksymalna wartość dla UNSIGNED CHAR = 255
Minimalna wartość dla CHAR = -128
Maksymalna wartość dla CHAR = 127

Minimalna wartość dla SHORT INT = -32768
Maksymalna wartość dla SHORT INT = 32767
Maksymalna wartość dla UNSIGNED SHORT INT = 65535
Minimalna wartość dla INT = -2147483648
Maksymalna wartość dla INT = 2147483647
Maksymalna wartość dla UNSIGNED INT = 4294967295
Minimalna wartość dla LONG INT = -2147483648
Maksymalna wartość dla LONG INT = 2147483647
Maksymalna wartość dla UNSIGNED LONG INT = 4294967295
Minimalna wartość dla LONG LONG INT = -9223372036854775808
Maksymalna wartość dla LONG LONG INT = 9223372036854775807
Maksymalna wartość dla UNSIGNED LONG LONG INT = 18446744073709551615

Maksymalna wielobajtowa długość znaku we wszystkich ustawieniach regionalnych = 5

-----
Process exited after 10.79 seconds with return value 0
Press any key to continue . . .

```

Jakie mogą być konsekwencje przekroczenia tych ograniczeń?
Poważne! Oto przykład:

```

// Przekroczenie zakresu zmiennej

#include <stdio.h>
#include <limits.h>           // ten temat
#include <locale.h>           // dla 'setlocale()'

int main()

```

```

{  setlocale(LC_CTYPE, "Polish");  // polskie znaki
   short int a = 30000;
   short int b = 40000;           // przekroczona wartość dla typu liczbowego

   printf("\n Maksymalna wartość dla SHORT INT = %d\n\n", SHRT_MAX);
   printf(" Drukuję wartość 'a' : %d\n", a);
   printf(" Drukuję wartość 'b' : %d\n", b);

   return 0;
}

```

Wynik działania programu:

```

Maksymalna wartość dla SHORT INT = 32767

Drukuję wartość 'a' : 30000
Drukuję wartość 'b' : -25536

-----
Process exited after 2.281 seconds with return value 0
Press any key to continue . . .

```

Zauważ 'return value = 0' co oznacza, że kompilator nie znalazł tu niczego złego!

Trudno doszukać się sensu uzyskanej liczby: Powstaje ona przez obcięcie liczby zbyt dużej. Znak 'minus' przy tej liczbie to efekt bitu '1' po jej najbardziej lewej stronie - wtedy komputer traktuje liczbę jak ujemną (patrz: "Tablice języka C" a w nim "two's complements").

Wniosek: Bądź ostrożny.