

Wykonał: Radosław Smoter

Grupa: 14

Nr: 27

Numer zadania: 4

Przykład: 62

Prowadzący: Prof. dr hab. inż.
Volodymyr Samotyy

Politechnika Krakowska

Wydział Inżynierii Elektrycznej i Komputerowej

Sprawozdanie: Wstęp do Programowania

Spis treści

Polecenie.....	1
Kod programu.....	2
Wyniki.....	3
Opis Programu.....	4
Wnioski.....	5

Polecenie

Obliczyć wartość funkcji jednoargumentowej. Z wartości y wybieramy dwie (trzy) cyfry i wyświetlamy wynik w systemach: dwójkowym, ósemkowym i szesnastkowym. Dla poszczególnych zadań wybieramy cyfry zgodnie z tabelą.

Zadanie	Cyfry
62	3, 5, 6

Funkcja: $y = \frac{5}{3 + \lg^2(x)}$.

Dziedzina: $x = 1.2$.

Kod programu

(1) wp_smoter_r_4.c

```
#include "handleConversions.h"

int main(void)
{
    double x = 1.2; // Value as domain

    extern double doMath(double);
    extern void showValues(double);

    double funcVal = doMath(x); // Math function result
    if (funcVal == NAN) return -1; // Handle NAN exception

    showValues(funcVal);

    return 0;
}
```

(2) handleConversions.h

```
#include "math.h"
#include "decToBin.h"

/*
 * Do a mathematical function
 * Parameter: value as domain
 */
double doMath(double x) {
    double log1 = log(x);
    if (isnan(log1)) return NAN;
    return 5 / (3 + pow(log1 / log(10), 2));
}

/*
 * Select k-th digit from number n
 * Parameters: number, digit number
 */
unsigned short int selectDigit(double n, int k) {
    // Format number into scientific notation
    // without maintaining the powerset
    // e.g. abc.def --> a.bcdef * 10^0 <=> a.bcdef
    while (n < 0)
        n *= 10;
    while (n > 10)
        n /= 10;

    // Transform from form a.bcdef into a.bcdef * 10^(k-1)
    // e.g. for the first digit it is a.bcdef * 10^0
    // for the fifth, a.bcdef * 10^4
    for (int i = 0; i < k - 1; i++)
        n *= 10;
}
```

```

    // Cast n into integer; return wanted digit
    // e.g. (a.bcdef * 10^0) % 10 = a
    // (a.bcdef * 10^4) % 10 = e
    return (int)n % 10;
}

/*
 * Show a value in various counting systems:
 * In bin, oct and hex
 * Parameter: value
 */
void showValues(double val) {

    // Number of digit to choose from val (from left)
    unsigned short int digitPlace[] = { 3, 5, 6 };
    // Chosen digits
    unsigned short int digits[3];

    // Choose digits from val
    for (int i = 0; i < 3; i++) {
        digits[i] = selectDigit(val, digitPlace[i]);
    }

    // Full value
    printf("%10s%20.12f\n", "Value:", val);

    // Counting system
    printf("%10s%10s%10s%10s\n", "bin", "oct", "dec", "hex");

    extern void decToBin(unsigned long int);

    for (int i = 0; i < 3; i++)
    {
        // As bin
        decToBin(digits[i]);
        // As oct
        printf("%10o", digits[i]);
        // As dec
        printf("%10i", digits[i]);
        // As hex
        printf("%10X", digits[i]);

        printf("\n");
    }
}

```

(3) decToBin.h

```

#include "stdio.h"

// Utility function
// Check if number is even
int isEven(long int n) { return n % 2; }

/*
 * Convert decimal integer to binary form
 * Integer must be positive
 * Parameters:
 * integer (dec)
 */
void decToBin(unsigned long int n)

```

```

{
    int remainder[65]; // Remainders of division (n % 2)
    int i = 0; // Current index

    do {
        remainder[i] = isEven(n);
        i++;
    } while ((n /= 2) > 0);

    // Determine how many zeros to add to remainder to maintain format in eights
    // with zeros in the beginning, if needed
    while(i % 8)
    {
        remainder[i] = 0;
        i++;
    }

    char bin[65]; // Binary form of n
    // Flip remainder
    // Add '0' to every integer so it becomes it's corresponding character
    for (int j = 0; j < i; j++)
    { bin[j] = remainder[i - j - 1] + '0'; }

    // End as string
    bin[i] = '\0';

    // Print formatted version onto the terminal
    printf("%10s", bin);
}

```

Wyniki

Value:	1.663190781037		
bin	oct	dec	hex
00000110	6	6	6
00000001	1	1	1
00001001	11	9	9

Opis Programu

(1) Funkcja doMath()

Prototyp:

```
double doMath(double)
```

Parametry:

- wartość jako dziedzina

Działanie:

Oblicza wartość funkcji matematycznej. Najpierw oblicza wartość logarytmu $\log(x)$. Jeśli x nie mieści się w dziedzinie logarytmu, to zwraca NAN . W innym przypadku oblicza i zwraca wartość wyrażenia.

Efekt:

Oblicza wartość działania matematycznego, $\frac{5}{3+\lg^2(x)}$, dla podanego parametru.

(2) Funkcja showValues()

Prototyp:

```
void showValues(double)
```

Parametry:

- Wartość

Działanie:

Pokazuje wartość z parametru w różnych systemach liczenia: dwójkowym, ósemkowym, dziesiętnym, szesnastkowym. Za pomocą funkcji selectDigit(), wybiera z parametru cyfry o pożądanym położeniu, licząc od lewej strony. Dalej, wypisuje w sformatowany sposób wartość całej liczby; wcześniej uzyskane cyfry w odpowiednich systemach liczbowych. Funkcją decToBin() w systemie binarnym; funkcjami printf() ze specyfikatorami "o" - system ósemkowy, "i" - dziesiętny, "X" - szesnastkowy.

Efekt:

Wypisanie 3-ciej, 5-tej, 6-tej cyfry podanego parametru w systemach dwójkowym, ósemkowym, dziesiętnym, szesnastkowym.

(3) Funkcja selectDigit()

Prototyp:

```
unsigned short int selectDigit(double, int)
```

Parametry:

- Liczba
- Numer cyfry

Działanie:

Najpierw, przesuwa miejsce dziesiętne dostarczonej liczby do postaci z notacji wykładniczej, nie zachowując potęgi (tj. $liczba \cdot 10^0$). Dalej przesuwa miejsce przecinka do k-tej pozycji mnożąc przez 10^{k-1} . Funkcja zwraca resztę z dzielenia takiej liczby przez 10.

Efekt:

Podanie k-tej cyfry liczby n.

(4) Funkcja isEven()

Prototyp:

```
int isEven(long int)
```

Parametry:

- Liczba całkowita

Działanie:

Zwraca resztę z dzielenia liczby n przez 2.

Efekt:

Określenie parzystości liczby.

(5) Funkcja decToBin()

Prototyp:

```
void decToBin(unsigned long int)
```

Parametry:

- Liczba całkowita

Działanie:

Dla każdego dodatniego n , od drugiej iteracji ($n \div 2$) zapisuje resztę z dzielenia do tablicy remainder. Dalej populuje remainder zerami nieznaczącymi (dla działań), do postaci wielokrotności liczby 8 ilości miejsc zajmowanych w tablicy. Później odwraca tablicę oraz dodaje

'0', które zamienia liczby z postaci $[0,1]$ do korespondującej postaci ASCII dla ich wartości znakowych ['0', '1']. Kończy tablicę przez '\0', by była przyjazna operacjom na napisach. Wypisuje sformatowaną wersję za pomocą printf().

Efekt:

Zamiana liczby na postać binarną i wypisanie jej to terminala.

Wnioski

Wyrażenie matematyczne jest poprawnie obliczone.

Value: 1.663190781037



5 / (3 + lg(x)^2) for 1.2

NATURAL LANGUAGE MATH INPUT

EXTENDED KEYBOARD EXAMPLES UPLOAD RANDOM

Assuming "lg" is the base 10 logarithm | Use [the natural logarithm](#) or [the base 2 logarithm](#) instead

Input interpretation

$$\frac{5}{3 + \log_{10}^2(x)}$$
 where $x = 1.2$

Result

1.66319

Na tej podstawie można stwierdzić, że przedstawione liczby również są poprawnie wybrane.