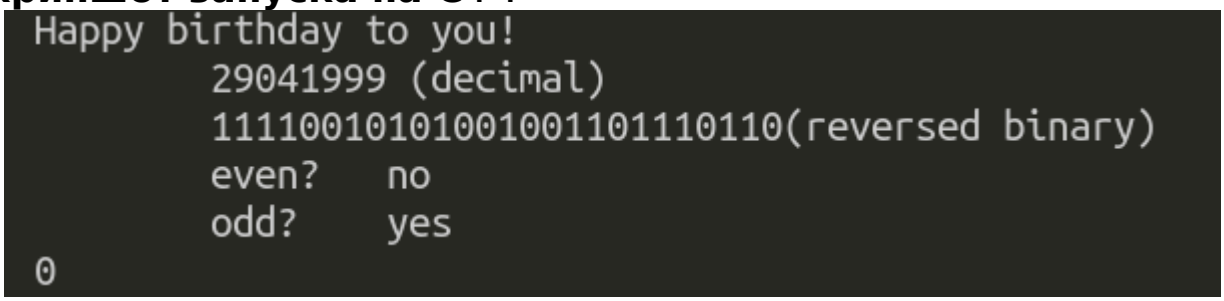
Правки:
Лабораторная работа N1

Распечатка файла even-odd.cpp

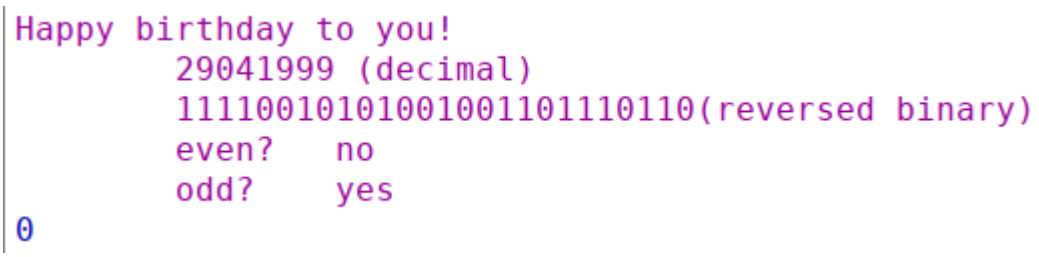
```
#include "mlisp.h"

double dd = 29;
double mm = 4;
double yyyy = 1999;

double even__bits(double n);
double odd__bits(double n);
double display__bin(double n);
double report__results(double n);

double even__bits(double n){
	return (n == 0 ? 1
		: (remainder(n, 2) == 0 ?
		even__bits(quotient(n, 2))
		: odd__bits(quotient(n, 2))));
}

double odd__bits(double n){
	return (n == 0 ? 0
		: remainder(n, 2) == 0 ?
		odd__bits(quotient(n, 2))
		: true ? even__bits(quotient(n, 2))
		: _infinity);
}

double display__bin(double n){
	display(remainder(n, 2));
	return n == 0 ? 0 : display__bin(quotient(n, 2));
}

double report__results(double n){
	display("Happy birthday to you!\n\t");
	display(n); display(" (decimal)\n\t");
	display__bin(n); display("(reversed binary)\n");
	display("\teven?\t"); display(even__bits(n) == 1 ? "yes" :
"no");
	newline();
```

```
	display("\todd?\t"); display(odd__bits(n) == 1 ? "yes" :
"no");
	newline();
	return 0;
}

int main(){
	display(report__results(dd * 1000000 +
		mm * 10000 +
		yyyy));
	newline();

	std::cin.get();
	return 0;
}
```

Скриншот запуска на C++



Скриншот запуска на Лиспе



Лабораторная работа N2

Распечатка файла golden-section.cpp

```
#include "mlisp.h"

double a = 0;
double b = 2;
double eps = 0.00001;
double mphi = 0;
double xmin = 0;

double fun(double x);
double golden__section__search(double a, double b);
```

```
double golden__start(double a, double b);
double __KAV__try(double a, double b, double xa, double ya,
double xb, double yb);

double fun(double x){
   x = x - double(109) / 110 / e;
   return 5 * expt(log(expt(atan(x - 2), 2)), 4) - x - 7;
}

double golden__section__search(double a, double b){
   {
      double xmin = a < b ? golden__start(a, b) :
golden__start(b, a);
      newline();
      return xmin;
   }
}

double golden__start(double a, double b){
   mphi = 0.5 * (3 - sqrt(5));
   {
      double xa = a + mphi * (b - a);
      double xb = b - mphi * (b - a);
      return __KAV__try(a, b, xa, fun(xa), xb, fun(xb));
   }
}

double __KAV__try(double a, double b, double xa, double ya,
double xb, double yb){
   return (abs(a - b) < eps ? (a + b) * 0.5
      : (true ?
         (display("+"), ya < yb ?
            b = xb,
            xb = xa,
            yb = ya,
            xa = a + mphi * (b - a),
            __KAV__try(a, b, xa, fun(xa), xb, yb)
            : (a = xa,
            xa = xb,
            ya = yb,
            xb = b - mphi * (b - a),
            __KAV__try(a, b, xa, ya, xb, fun(xb))))
      : _infinity));
}
```

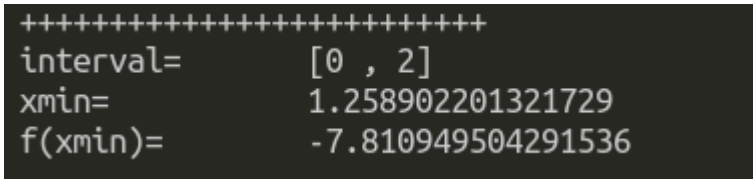
```
int main(){
   xmin = golden__section__search(a, b);
   display("interval=\t[");
   display(a);
   display(" , ");
   display(b);
   display("]\n");
   display("xmin=\t\t");
   display(xmin); newline();
   display("f(xmin)=\t");
   display(fun(xmin)); newline();
   std::cin.get();
   return 0;
}
```

Распечатка файла golden-section.ss

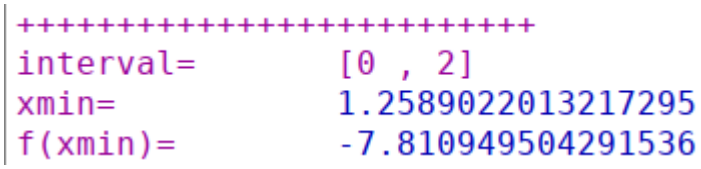
```
;golden-section
(define a 0)(define b 2)
(define z 0)
(define (fun x)
 (set! x (- x (/ 109 110 e)))
 (set! z  x)
 (- (* 5 (expt (log (expt (atan (- z 2)) 2)) 4)) z 7)
;     5*ln^4(arctg^2(z-2))                    - z - 7
)
(define eps 0.00001)
(define (golden-section-search a b)
 (let(
      (xmin(if(< a b)(golden-start a b)(golden-start b a )))
     )
     (newline)
     xmin
 )
)
(define (golden-start a b)
 (set! mphi(* 0.5(- 3(sqrt 5))))
 (let(
      (xa (+ a (* mphi(- b a))))
      (xb (- b (* mphi(- b a))))
     )
     (try a b xa (fun xa) xb (fun xb))
 )
)
```

```
(define mphi 0)
(define (try a b xa ya xb yb)
 (cond((<(abs (- a b))eps) (* (+ a b)0.5))
     (#t (display "+")
        (cond((< ya yb)(set! b xb)
                    (set! xb xa)
                    (set! yb ya)
                    (set! xa (+ a (* mphi(- b a))))
                    (try a b xa (fun xa) xb yb)
              )
              (else   (set! a xa)
                    (set! xa xb)
                    (set! ya yb)
                    (set! xb (- b (* mphi(- b a))))
                    (try a b xa ya xb (fun xb))
              )
        )
      )
 )
)
(define xmin 0)
(set! xmin(golden-section-search a b))
  (display"interval=\t[")
  (display a)
  (display" , ")
  (display b)
  (display"]\n")
  (display"xmin=\t\t")
xmin
  (display"f(xmin)=\t")
(fun xmin)
```

Скриншот запуска на C++



Скриншот запуска на Лиспе



Лабораторная работа N3

Распечатка файла coin19.cpp

```
//coin19.cpp 2019
#include "mlisp.h"

double VARIANT = 9;
double LAST__DIGIT__OF__GROUP__NUMBER = 8; //6,8
double LARGEST__COIN = 20;

bool implication_Q(bool x_Q, bool y_Q);
double cc(double amount, double largest__coin);
double count__change(double amount);
double next__coin(double coin);
double GR__AMOUNT();

bool implication_Q(bool x_Q, bool y_Q){
   return !(x_Q && !y_Q);
}

double cc(double amount, double largest__coin){
   return ((amount == 0 || largest__coin == 1) ? 1
   : implication_Q(amount >= 0, largest__coin == 0) ? 0
   : (cc(amount, next__coin(largest__coin)) +
   cc(amount - largest__coin, largest__coin)));
}

double count__change(double amount){
   return cc(amount, LARGEST__COIN);
}

double next__coin(double coin){
   return (coin == 20 ? 15
   : coin == 15 ? 10
   : coin == 10 ? 5
   : coin == 5 ? 3
   : coin == 3 ? 2
   : 1);
}

double GR__AMOUNT(){
   return remainder(100 *
LAST__DIGIT__OF__GROUP__NUMBER + VARIANT, 137);
}
```

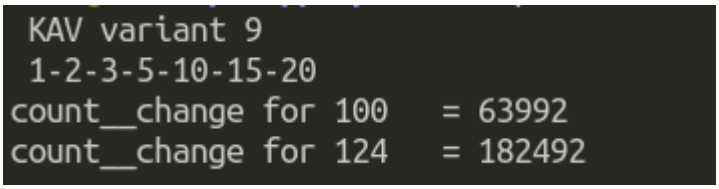
```
int main(){
   display(" KAV variant ");
   display(VARIANT); newline();
   display(" 1-2-3-5-10-15-20"); newline();
   display("count__change for 100 \t= ");
   display(count__change(100)); newline();
   display("count__change for ");
   display(GR__AMOUNT());
   display(" \t= ");
   display(count__change(GR__AMOUNT())); newline();

   std::cin.get();
   return 0;
}
```

Распечатка файла coin19.ss

```
(define VARIANT 9)
(define LAST-DIGIT-OF-GROUP-NUMBER 8)
(define LARGEST-COIN 20)

(define (implication? x? y?) (not (and x? (not y?))))

(define (cc amount largest-coin) (cond
            ((or (= amount 0) (= largest-coin 1)) 1)
            ((implication? (>= amount 0) (= largest-coin 0)) 0)
            (else (+ (cc amount (next-coin largest-coin))
                  (cc (- amount largest-coin) largest-coin))) )
)

(define (count-change amount) (cc amount LARGEST-COIN))

(define (next-coin coin) (cond ((= coin 20) 15)
                     ((= coin 15) 10)
                     ((= coin 10) 5)
                     ((= coin 5) 3)
                     ((= coin 3) 2)
                     (else 1) )
)

(define (GR-AMOUNT) (remainder (+ (* 100 LAST-DIGIT-OF-
GROUP-NUMBER) VARIANT) 137))

(display " KAV variant ")
```

(display VARIANT) (newline)
(display " 1-2-3-5-10-15-20") (newline)
(display "count_change for 100 \t= ")
(display (count-change 100)) (newline)
(display "count_change for ");
(display (GR-AMOUNT))
(display " \t= ")
(display (count-change (GR-AMOUNT)))(newline)

Скриншот запуска на C++



Скриншот запуска на Лиспе