

Вариант 2 (условия и решения)

Задача 1 (8 баллов). Перевести шестнадцатеричное число $A_{16} = 15D,CC$ в десятичную систему счисления.

Решение задачи 1.

- 1) $15D = 1 \cdot 16^2 + 5 \cdot 16^1 + 13 \cdot 16^0 = 1 \cdot 256 + 5 \cdot 16 + 13 \cdot 1 = 256 + 80 + 13 = 349_{10}$.
- 2) $0,CC = 12 \cdot 16^{-1} + 12 \cdot 16^{-2} = 12 \cdot (1/16) + 12 \cdot (1/16/16) = 3 \cdot (1/4) + 3 \cdot (1/4/16) = (3/4) + (3/64) = 0,75 + 0,046875 = 0,796875_{10}$.

Ответ: $A_{10} = 349,796875$.

Задача 2 (8 баллов). Найти сумму шестнадцатеричных чисел $A_{16} = 667$ и $B_{16} = F56$, используя 16-разрядный сумматор, старший разряд которого знаковый. Ответ дать в шестнадцатеричной форме. При переполнении разрядной сетки ответ сопроводить сообщением.

Решение задачи 2.

- 1) $[A_2]_{пр} = 0,000\ 0110\ 0110\ 0111$ $[B_2]_{пр} = 0,000\ 1111\ 0101\ 0110$
- 2) $[A_2]_{пр}^M = 00,000\ 0110\ 0110\ 0111$ $[B_2]_{пр}^M = 00,000\ 1111\ 0101\ 0110$
- 3) $[A_2]_{пр}^M + [B_2]_{пр}^M = 00,000\ 0110\ 0110\ 0111 + 00,000\ 1111\ 0101\ 0110 = 00,001\ 0101\ 1011\ 1101$
- 4) $[C_2]_{пр} = 0,001\ 0101\ 1011\ 1101$

Ответ: $C_{16} = 15BD$.

Задача 3 (8 баллов). Дано выражение, в котором используются операции над булевскими величинами, принимающими значения 0 (ложь) и 1 (истина). Выражение может содержать круглые скобки и следующие знаки операций: отрицание (\neg), конъюнкция (\wedge), дизъюнкция (\vee), импликация (\rightarrow). Операции имеют следующие уровни приоритета: уровень 1 (\neg), уровень 2 (\wedge), уровень 3 (\vee), уровень 4 (\rightarrow). Построить таблицу истинности для выражения $\neg((p \rightarrow q) \wedge \neg r) \rightarrow (p \vee \neg r)$.

Решение задачи 3.

p	q	r	$p \rightarrow q$	$\neg r$	$(p \rightarrow q) \wedge \neg r$	$\neg((p \rightarrow q) \wedge \neg r)$	$p \vee \neg r$	$\neg((p \rightarrow q) \wedge \neg r) \rightarrow (p \vee \neg r)$
0	0	0	1	1	1	0	1	1
0	0	1	1	0	0	1	0	0
0	1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	0	0
1	0	0	0	1	0	1	1	1
1	0	1	0	0	0	1	1	1
1	1	0	1	1	1	0	1	1
1	1	1	1	0	0	1	1	1

Задача 4 (8 баллов). Дано выражение, в котором используются поразрядные операции над 8-ми разрядными целыми числами без знака. В выражении используются круглые скобки и следующие знаки операций: поразрядное НЕ (\sim), поразрядное И ($\&$), поразрядное ИЛИ (\mid), поразрядный сдвиг влево (\ll), поразрядный сдвиг вправо (\gg). Операции имеют следующие уровни приоритета: уровень 1 (\sim), уровень 2 (\ll и \gg), уровень 3 ($\&$), уровень 4 (\mid). Вычислить значение следующего выражения: $a \gg 2 \& b \ll 1 \mid \sim(a \ll 2 \& b \gg 1)$ для $a = 240$ и $b = 63$. Ответ дать в двоичной и десятичной формах.

Решение задачи 4.

- 1) $a = f0_{16} = 11110000_2$
- 2) $b = 3f_{16} = 00111111_2$
- 3) $a \gg 2 = 3c_{16} = 00111100_2$
- 4) $b \ll 1 = 7e_{16} = 01111110_2$
- 5) $a \gg 2 \& b \ll 1 = 3c_{16} = 00111100_2$
- 6) $a \ll 2 = c0_{16} = 11000000_2$
- 7) $b \gg 1 = 1f_{16} = 00011111_2$
- 8) $a \ll 2 \& b \gg 1 = 0_{16} = 00000000_2$
- 9) $\sim(a \ll 2 \& b \gg 1) = ff_{16} = 11111111_2$
- 10) $a \gg 2 \& b \ll 1 \mid \sim(a \ll 2 \& b \gg 1) = ff_{16} = 11111111_2$

Ответ: $11111111_2 = 255_{10}$.

Задача 5 (8 баллов). Пусть $\{a_n\}$ и $\{b_n\}$ ($n \geq 1$) – две последовательности, члены которых связаны соотношением $a_{n+1} = 3a_n + b_n$, $b_{n+1} = -a_n + b_n$. Вычислить a_5 и b_5 , зная, что $a_1 = 14$, $b_1 = -6$.

Решение задачи 5.

Первый способ. $a_1 = 14$, $b_1 = -6$; $a_2 = 36$, $b_2 = -20$; $a_3 = 88$, $b_3 = -56$; $a_4 = 208$, $b_4 = -144$; $a_5 = 480$, $b_5 = -352$.

Второй способ. Можно найти выражения для a_n и b_n через n , считая, что a_1 и b_1 заданы. Выражения будут такими: $a_n = (5 + 2 \cdot n) \cdot 2^n$ и $b_n = -(2 \cdot n + 1) \cdot 2^n$. Для $n = 5$ будем иметь: $a_5 = (5 + 2 \cdot 5) \cdot 2^5 = 15 \cdot 32 = 480$, $b_5 = -(2 \cdot 5 + 1) \cdot 2^5 = -11 \cdot 32 = -352$.

Ответ: $a_5 = 480$, $b_5 = -352$.

Задача 6 (8 баллов). Дана инфиксная запись арифметического выражения $(a + b) * (((e - f) + g) / h) - i$. Найти префиксную запись этого выражения.

Решение задачи 6.

Сначала надо построить бинарное дерево, изображающее арифметическое выражение. Корень дерева и все внутренние вершины дерева являются бинарными операторами, а листья дерева – операндами. Затем обойти дерево в прямом порядке (pre-ordered): вершина – левое поддерево – правое поддерево.

Ответ: $* + a b - / + - e f g h i$.

Задача 7 (8 баллов). Рассмотрим шахматную доску со стороной 6. Требуется провести ладью из левого нижнего угла в правый верхний угол. Двигаться можно только вверх и вправо, не заходя при этом на клетки диагонали, соединяющей начальную и конечную клетки, и ниже нее. (Ладья оказывается на этой диагонали только в начальный и конечный моменты времени.) Сколько существует таких маршрутов?

Решение задачи 7.

Первый способ. Задачу можно решить методом полного перебора.

Второй способ. Можно решить задачу методом рекурсивного подсчета. В каждой клетке шахматной доски 6×6 напомним число, равное количеству допустимых путей, которыми ладья может прийти до этой клетки из левого нижнего угла.

- Заполняем нулями все запрещенные клетки.
- Начиная с левого нижнего угла, заполняем единицами всю левую вертикаль.
- Далее заполняем вторую слева вертикаль по следующему правилу: в очередной клетке ставим сумму чисел, стоящих в двух соседних клетках – снизу и слева, и так далее.

Таким образом, мы находим, что искомое число маршрутов равно 14.

1	4	9	14	14	14
1	3	5	5	0	0
1	2	2	0	0	0
1	1	0	0	0	0
1	0	0	0	0	0
1	0	0	0	0	0

Третий способ. Можно показать, что число маршрутов ладьи, удовлетворяющих условию задачи, определяется четвертым числом Каталана. Само число Каталана выражается формулой $C(n) = (2n)!/n!/(n+1)!$. $C(4) = (2 \cdot 4)!/4!/(4+1)! = (8)!/4!/(5)! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8 / (1 \cdot 2 \cdot 3 \cdot 4 \cdot 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5) = 5 \cdot 6 \cdot 7 \cdot 8 / (1 \cdot 2 \cdot 3 \cdot 4 \cdot 5) = 7 \cdot 2 = 14$.

Ответ: 14.

Задача 8 (12 баллов). Сколько существует положительных целых чисел между 1 и 2003, которые

- Делятся на 5?
- Делятся на 7?
- Делятся на 11?
- Делятся на 5 или 7 или 11?

Решение задачи 8.

$$\left[\frac{2003}{5} \right] = 400$$

Имеется всего _____ целых чисел, которые делятся на 5.

$\lfloor \frac{2003}{7} \rfloor = 286$
 Имеется всего 286 целых чисел, которые делятся на 7.

$\lfloor \frac{2003}{11} \rfloor = 182$
 Имеется всего 182 целых числа, которые делятся на 11.

$\lfloor \frac{2003}{5 \cdot 7} \rfloor = 57$
 Имеется всего 57 целых чисел, которые делятся на 5 и 7.

$\lfloor \frac{2003}{7 \cdot 11} \rfloor = 26$
 Имеется всего 26 целых чисел, которые делятся на 7 и 11.

$\lfloor \frac{2003}{11 \cdot 5} \rfloor = 36$
 Имеется всего 36 целых чисел, которые делятся на 11 и 5.

$\lfloor \frac{2003}{5 \cdot 7 \cdot 11} \rfloor = 5$
 Имеется всего 5 целых чисел, которые делятся на 5, 7 и 11.

Следовательно, количество целых чисел, которые делятся или на 5, или на 7, или на 11 равно $400 + 286 + 182 - 57 - 26 - 36 + 5 = 754$.

Ответ: а) 400; б) 286; в) 182; г) 754.

Задача 9 (16 баллов). Островом в последовательности целых чисел называется подпоследовательность смежных чисел, в которой каждое число больше числа, находящегося непосредственно слева и справа от подпоследовательности. Например, в последовательности 0 0 1 1 2 2 1 1 0 1 2 0 имеется 4 острова, которые выделены скобками: 0 0 (1 1 (2 2) 1 1) 0 (1 (2)) 0. Найти количество островов в последовательности.

Входные данные. Каждая строка входного файла содержит последовательность из 12 неотрицательных целых чисел, каждое из которых не больше 10^6 . Первое и последнее число последовательности равны 0.

Выходные данные. В выходной файл для каждой строки входного файла вывести одно целое число – количество островов в последовательности.

Пример входных данных	Пример выходных данных
0 0 1 1 2 2 1 1 0 1 2 0	4
0 1 2 4 4 1 0 2 4 1 0 0	6

Решение задачи 9.

Язык Си.

```

#include "stdafx.h"
int const N_TERMS = 12;
FILE *ifs, *ofs;
int _tmain(int argc, _TCHAR* argv[]) {
    int seq[N_TERMS], lev[N_TERMS+1], level, prev, curr, nisland;
    if( (ifs = fopen( "input.txt", "r" )) == NULL ) return 1;
    if( (ofs = fopen( "output.txt", "w" )) == NULL ) return 2;
    while (true) {
        level = prev = nisland = 0;
        fscanf( ifs, "%d", &curr);
        if ( feof( ifs ) ) break;
        lev[0] = prev = curr;
        for(int i = 1; i < N_TERMS ; i++) {
            fscanf( ifs, "%d", &curr);
            seq[i] = curr;
            if(curr > prev) { level++; nisland++; lev[level] = curr; }
            else if(curr < prev) {
                while((level > 0) && (curr < lev[level])) level--;
                if(curr > lev[level]) { level++; nisland++; lev[level] = curr; }
                prev = curr;
            }
        }
        fprintf( ofs, "%d\n", nisland);
    }
    return 0;
}

```

Язык Паскаль.

```

program Project_2_9;
{$APPTYPE CONSOLE}
uses SysUtils;
const N_TERMS = 12;
var f, g: TextFile;
    seq: array[0..N_TERMS-1] of integer;
    lev: array[0..N_TERMS] of integer;
    level, prev, curr, nisland, i: integer;
begin
  AssignFile(f, 'input.txt');
  AssignFile(g, 'output.txt');
  try
    Reset(f);
    Rewrite(g);
    try
      nisland := 0;
      while (not Eof(f)) do
        begin
          nisland := 0; prev := 0; level := 0;
          Read(f, curr);
          lev[0] := prev;
          prev := curr;
          for i := 1 to N_TERMS-1 do
            begin
              Read(f, curr);
              seq[i] := curr;
              if (curr > prev) then
                begin
                  Inc(level); Inc(nisland); lev[level] := curr;
                end
              else if (curr < prev) then
                begin
                  while((level > 0) and (curr < lev[level])) do Dec(level);
                  if (curr > lev[level]) then
                    begin
                      Inc(level); Inc(nisland); lev[level] := curr;
                    end
                  end;
                end;
              prev := curr;
            end;
          Readln(f);
        end;
        Writeln(g, nisland);
      finally
        CloseFile(f);
        CloseFile(g);
      end;
    except
      on EInOutError do Writeln('EInOutError!');
    end;
  end.
end.

```

Задача 10 (16 баллов). Ортогональную целочисленную решетку, состоящую из точек с целыми координатами в декартовой системе координат, будем обозначать через Z^2 . В узлах решетки Z^2 заданы две точки A и B. Требуется перечислить координаты левых нижних углов всех единичных квадратов решетки Z^2 , которые пересекаются с отрезком AB.

Входные данные. Во входном файле заданы целые числа Ax, Ay, Bx, By – координаты точек A и B. Все числа по абсолютной величине не превосходят 1000. Отрезок AB не параллелен осям координат.

Выходные данные. В выходной файл вывести координаты левых нижних углов единичных квадратов, которые имеют хотя бы одну общую точку с отрезком AB. Порядок перечисления квадратов может быть произвольным.

Пример входных данных	Пример выходных данных
-1 -1 1 2	[-2, -2] [-1, -2] [-2, -1] [-1, -1] [-1, 0] [0, 0] [0, 1] [0, 2] [1, 1] [1, 2]

Решение задачи 10.

Язык Си.

```

#include "stdafx.h"
#include <stdlib.h>

```

```

FILE *ifs, *ofs;
int Ax, Ay, Bx, By;
void swap ( int &x, int &y ) { int temp = x; x = y; y = temp; }
void print_result( int x, int y ) { fprintf( ofs, "[%d, %d] ", x, y ); }
void solve_up() {
    int x = Ax; int y = Ay; int d = (By - Ay) - (Bx - Ax);
    print_result( x-1, y-1 );
    print_result( x, y-1);
    print_result( x-1, y );
    print_result( x, y );
    for (int i = 1; i <= abs(By - Ay + Bx - Ax); i++) {
        if (d == 0) print_result( x, y+1 );
        if (d > 0) { y++; d -= Bx - Ax; }
        else { x++; d += By - Ay; }
        print_result( x, y );
    }
}
void solve_down() {
    int x = Ax; int y = Ay; int d = (Ay - By) - (Bx - Ax);
    print_result( x-1, y-1 );
    print_result( x, y-1 );
    print_result( x-1, y);
    print_result( x, y );
    for (int i = 1; i <= abs( Ay-By+Bx-Ax ); i++) {
        if (d == 0) print_result( x, y-2 );
        if (d > 0) { y++; d -= Bx - Ax; }
        else { x++; d += Ay - By; };
        print_result( x, y-1 );
    }
}
void solve_vert() {
    for (int i = Ay - 1; i <= By; i++) { print_result( Ax-1, i ); print_result( Ax, i ); }
}
void solve_horiz() {
    for (int i = Ax - 1; i <= Bx; i++) { print_result( i, Ay-1 ); print_result( i, Ay ); }
}
int _tmain(int argc, _TCHAR* argv[]) {
    if( (ifs = fopen( "input.txt", "r" )) == NULL ) return 1;
    if( (ofs = fopen( "output.txt", "w" )) == NULL ) return 2;
    fscanf( ifs, "%d %d %d %d", &Ax, &Ay, &Bx, &By );
    if (Ax > Bx) { swap(Ax, Bx); swap(Ay, By); }
    else if (Ax == Bx) { if (Ay > By) swap( Ay, By ); }
    if (Ax == Bx) solve_vert();
    else if (Ay == By) solve_horiz();
    else if (By > Ay) solve_up();
    else solve_down();
    fclose(ifs);
    fclose(ofs);
    return 0;
}

```

Язык Паскаль.

```

program Project_2_10;
{$APPTYPE CONSOLE}
uses SysUtils;
var f, g: textfile;
    X1, Y1, X2, Y2, T: integer;
procedure PrintResult(X, Y: integer);
begin
    Write(g, '[' , X, ' ', Y, ' ] ');
end;
procedure SolveUp;
var D, X, Y, i: integer;
begin
    X := X1; Y := Y1; D := (Y2-Y1)-(X2-X1);
    PrintResult(X-1, Y-1);
    PrintResult(X, Y-1);
    PrintResult(X-1, Y);
    PrintResult(X, Y);

```

```

for i := 1 to abs(Y2 - Y1 + X2 - X1) do
begin
  if (D = 0) then PrintResult(X, Y+1);
  if (D > 0) then begin inc(Y); D := D - (X2 - X1); end
  else begin inc(X); D := D + (Y2 - Y1); end;
  PrintResult(X, Y);
end;
end;
procedure SolveDown;
var D, X, Y, i: integer;
begin
  X := X1; Y := Y1; D := (Y1-Y2)-(X2-X1);
  PrintResult(X-1, Y-1);
  PrintResult(X, Y-1);
  PrintResult(X-1, Y);
  PrintResult(X, Y);
  for i := 1 to Abs(Y1 - Y2 + X2 - X1) do
  begin
    if (D = 0) then PrintResult(X, Y-2);
    if (D > 0) then begin dec(Y); D := D - (X2 - X1); end
    else begin inc(X); D := D + (Y1 - Y2); end;
    PrintResult(X, Y-1);
  end;
end;
end;
procedure SolveVert;
var i: integer;
begin
  for i := Y1-1 to Y2 do begin PrintResult(X1-1, I); PrintResult(X1, I); end;
end;
procedure SolveHoriz;
var i: integer;
begin
  for i := X1-1 to X2 do begin PrintResult(I, Y1-1); PrintResult(I, Y1); end;
end;
end;
procedure Solve;
begin
  if (X1 = X2) then SolveVert
  else if (Y1 = Y2) then SolveHoriz
  else if (Y2 > Y1) then SolveUp
  else SolveDown;
end;
begin
  AssignFile(f, 'input.txt');
  AssignFile(g, 'output.txt');
  try
    Reset(f);
    Rewrite(g);
    try
      Readln(f, X1, Y1, X2, Y2);
      if (X1 > X2) then
        begin
          T := X1; X1 := X2; X2 := T;
          T := Y1; Y1 := Y2; Y2 := T;
        end
      else if (X1 = X2) then
        if (Y1 > Y2) then begin T := Y1; Y1 := Y2; Y2 := T; end;
      Solve;
      Writeln(g);
    finally
      CloseFile(f);
      CloseFile(g);
    end;
  except
    on EInOutError do Writeln('EInOutError!');
  end;
end.

```