

**Первый (заочный) этап академического соревнования
Олимпиады школьников «Шаг в будущее» по общеобразовательному предмету
«Информатика», осень 2015 г.
10 КЛАСС**

Задача 1. Перевести десятичное число 61,6875 в шестнадцатеричную систему счисления.

(4 баллов)

Решение

Перевод целой части достаточно очевиден. $61 = 3 \cdot 16 + 13 = 3D_{16}$

Для перевода дробной части.

$$0,6875 \cdot 16 = 11,0 = B.$$

Ответ: 3D,B

Задача 2. Используя 16-ти разрядный сумматор, старший разряд которого знаковый, вычислить сумму: $7C0F_{16}$ и 850_{16} . Результат запишите в 16-й система счисления. Числа со знаком, выражаемые с использованием 16 двоичных разрядов, должны находиться между -32768 и 32767 . При переполнении разрядной сетки ответ сопроводить сообщением.

(7 баллов)

Решение

Прямой код числа $7C0F_{16}$ равен 0111 1100 0000 1111 и, т.к. число положительно, мы имеем такой же дополнительный код.

Прямой код числа 850_{16} равен 0000 1000 0101 0000 и, т.к. число положительно, мы имеем такой же дополнительный код.

При сложении, из-за переполнения, получаем результат 1000 0100 0101 1111, который интерпретируется как дополнительный код отрицательного числа. Затем переводим в прямой код, который переводим в 16-ю систему счисления.

Ответ: -7BA1₁₆

Задача 3. Пусть $\{a_n\}$ ($n \geq 1$) – последовательность, для которой $a_{n+2} - 5a_{n+1} + 6a_n = 0$. Вычислить a_{10} , зная, что $a_1 = 3$, $a_2 = 7$.

(9 баллов)

Решение

$$a_{n+2} = 5a_{n+1} - 6a_n$$

a_1	3
a_2	7
a_3	17
a_4	43
a_5	113
a_6	307

a₇	857
a₈	2443
a₉	7073
a₁₀	20707

способ 2.

Составим характеристическое уравнение рекуррентной последовательности $a_{n+2} - 5a_{n+1} + 6a_n = 0$.

$r^2 - 5r + 6 = 0$. Корни уравнения, очевидно, $r_1 = 2$, $r_2 = 3$.

Составляем систему уравнений.

$$C_1 + C_2 = 3$$

$$2C_1 + 3C_2 = 7$$

Очевидно, $C_1 = 2$, $C_2 = 1$.

$$a_n = C_1 r_1^{n-1} + C_2 r_2^{n-1}; a_{10} = 2 \cdot 2^9 + 3^9 = 1024 + 19683 = 20707$$

Ответ: 20707

Задача 4. В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для обозначения логической операции «И» – символ «&».

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет:

<i>Запрос</i>	<i>Количество страниц (тыс.)</i>
Шайба	35
Клюшка	25
Мяч	40
Шайба Клюшка Мяч	70
Шайба & Мяч	10
Шайба & Клюшка	0

Сколько страниц (в тысячах) будет найдено по запросу

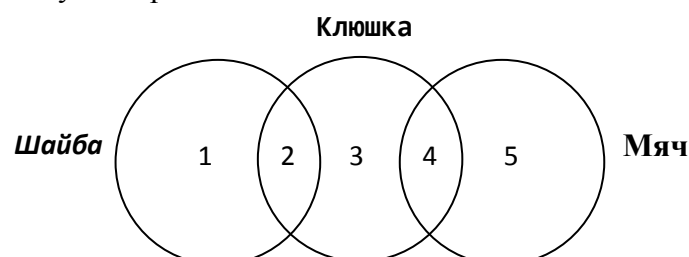
Клюшка & Мяч

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

(9 баллов)

Решение

1) построим диаграмму Эйлера-Венна



2) количество сайтов, удовлетворяющих запросу в области i , будем обозначать через N_i

3) здесь 5 областей, причём известны следующие данные:

$$N_1 + N_2 = 35$$

$$N_4 + N_5 = 25$$

$$N_2 + N_3 + N_4 = 40$$

$$N_1 + N_2 + N_3 + N_4 + N_5 = 70$$

$$N_2 = 10$$

4) нас интересует область 4. Находим ответ прямой подстановкой:

$$N_1 = 35 - N_2 = 25$$

$$N_5 = 70 - N_1 - (N_2 + N_3 + N_4) = 70 - 25 - 40 = 5$$

$$N_4 = 25 - N_5 = 20$$

5) таким образом, **ответ – 20.**

Задача 5.

В результате построения таблицы истинности логической формулы F получилась следующая таблица истинности

x	y	z	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Запишите эту формулу, используя минимальное количество логических операций. 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).

(11 баллов)

Решение

Заметим, что значение функции не зависит от x (строки 1-4 и 5-8 соответственно одинаковы). Далее рассматривая для переменных y и z строки 1-4 получаем таблицу истинности для операции импликации.

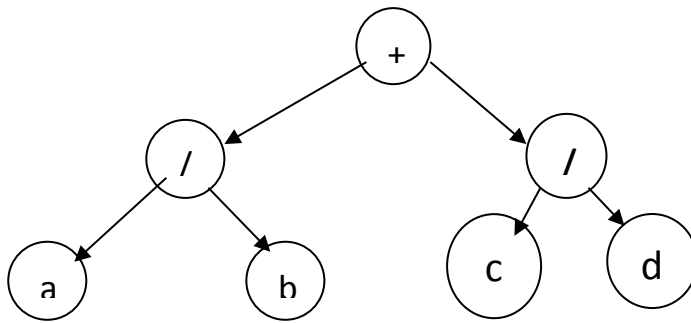
Ответ: $(\neg y) \vee z = y \rightarrow z$

Задача 6. Заменить инфиксное арифметическое выражение $a/b+c/d$ префиксным.

(11 баллов)

Решение

Представим выражение в виде дерева.



Ответ: +/ac/cd

Задача 7. На столе стоит блюдо со стопкой блинов, смазанных маслом. Диаметры всех блинов различны. С помощью лопатки Вы можете разделить стопку блинов на 2 части. Верхнюю часть можно перевернуть и положить обратно на нижнюю.

- Распишите Ваши действия, которые позволят расположить блины в стопке в порядке убывания их диаметров (постарайтесь сделать это эффективно).
- Оцените число требуемых переворотов.

(14 баллов)

Решение

Находим блин максимального диаметра, подсовываем под него лопатку и переворачиваем. Блин максимального диаметра оказывается сверху стопки блинов. Переворачиваем всю стопку. Блин с максимальным диаметром оказывается в самом низу, то есть на месте. Среди оставшихся блинов находим блин с максимальным диаметром, подсовываем под него лопатку и переворачиваем. Он оказывается сверху стопки блинов. Переворачиваем верхнюю часть стопки таким образом, что этот блин ложится на блин с максимальным диаметром. После этих действий упорядочены два блина. Продолжаем эти действия с оставшимися блинами, пока не останется два блина. Если последние два блина лежат в нужном порядке, то ничего делать не надо, если же они лежат неправильно, то их надо перевернуть.

Оценим количество переворотов. На каждый блин (кроме двух последних) в худшем случае мы затратим два переворота, то есть всего $2*(N-2)=2*N-4$ переворотов, где N количество блинов. Если последние два блина лежат не по порядку, то делаем ещё один переворот, то есть всего $2*N-4+1=2*N-3$ переворотов. Переворотов может быть меньше, если на каком-то шаге очередной блин уже будет лежать сверху стопки или, более того, на месте.

Задача 8. Обозначим через $C(n,k)$ количество сочетаний по k элементов из n . Обозначим через $P(n)$ количество перестановок n элементов. Написать программу для вычисления следующего выражения: $C(n,P(k))$.

Входные данные: Стандартный входной поток содержит строку, в которой записаны два целых неотрицательных числа n, k ($n \leq 10, k \leq 5$)

Выходные данные: В стандартный выходной поток вывести одно целое число – значение вычисленного выражения.

Пример входных данных	Пример выходных данных
3 2	3

(15 баллов)

Решение

program my;

function fact(x:integer):longint;

```

var i:integer;
    p:longint;
begin
p:=1;
for i:=2 to x do
    p:=i*p;
fact:=p;
end;

```

```

var n,k:integer;
begin
readln(n,k);
if fact(k)<=n then
    writeln(fact(n)/(fact(n-fact(k))*fact(fact(k))))
else
    writeln(0);
end.

```

Задача 9. Читальный зал школьной библиотеки часто посещается учениками на переменах. Для того, чтобы легко было найти среди множества книг нужную, они упорядочены по алфавиту. Ученики часто зачитываются книгами до самого звонка и, к сожалению, не всегда ставят их на нужное место. В итоге, после перемены книги стоят на полке совсем не по алфавиту и библиотекарю приходится тратить много времени для того, чтобы все привести в порядок.

Ребята пожалели библиотекаря и на кружке юных электронщиков создали робота-помощника. Так как они ещё только учились делать серьёзные вещи, то робот умел делать далеко не все, что хотелось бы, а точнее только следующие действия:

- прочитать и запомнить названия книг и их расположение;
- брать сразу две или три рядом стоящие книги и ставить их в том же порядке в любое место на полке (брать одну книгу или более трех он не может).

Кроме того, в школе не было кружка программистов и некому было написать программу для робота. Помогите ребятам и напишите программу, которая позволит расставить все книги по алфавиту. Исходные данные находятся в текстовом файле **INPUT.TXT**. Данные представлены следующим образом:

В первой строке файла записано число **n** ($3 < n < 500$)- количество книг на полке;

Затем следует **n** строк, в каждой из которых записано название одной книги (длина названия книги не превышает 200 букв, все названия на русском языке). Названия книг в файле расположены в том порядке, в котором книги стоят на полке. Очевидно, что в читальном зале может быть по несколько книг каждого названия.

Программа должна выдать в текстовый файл **OUTPUT.TXT** последовательность действий робота (если возможно несколько вариантов решения, то привести любой из них). Каждое действие робота записывается в отдельной строке файла в виде тройки чисел **x y z**, отделенных друг от друга пробелом. Такая запись означает, что робот должен взять книги начиная с номера **x** и заканчивая номером **y** и поставить их после книги с номером **z**. Здесь под номером книги подразумевается какой по счету от начала стоит книга. Для того, чтобы робот поставил книги в начало полки, необходимо указать **z**, равное нулю. При определении номера книги **z**, после которой необходимо вставить взятые книги, учитываются только те книги, которые в текущий момент стоят на полке (то есть без учета взятых роботом).

Число действий робота не должно превышать $2*n$.

Пример:

Пример входных данных	Пример выходных данных
6	3 5 0
Война и мир	1 2 2
Цель жизни	
Покушение на власть	
Приключения Оливера Твиста	

Решение

Общая идея решения.

Находим книгу, которая идёт раньше других по алфавиту, берём её и следующую за ней и переставляем на первое место. Среди оставшихся ищем ту, которая идёт раньше других по алфавиту, берём её и следующую за ней и переставляем на второе место и так далее. Имеется два особых случая. Первый – это когда искомая книга стоит в самом конце и за ней не стоит ни одной книги, то есть мы не можем взять две или три книги. В этом случае берём две книги, стоящие перед ней и ставим сразу после неё. Далее продолжаем в соответствии с описанным алгоритмом. Второй особый случай – это когда неупорядоченными остаются только две книги. В этом случае необходимо взять не две предпоследние книги, а три и перенести их в конец полки, а затем две предпоследние взять и поставить на одну позицию в сторону начала. Поясним на примере. Рассмотрим четыре последних книги. Две первых из них упорядочены, а две последних нет. Присвоим им номера от 1 до 4. Получим 1243 (две последних стоят не по порядку). Выполняем предложенный алгоритм. Берём 124 и переносим в конец, получаем 3124. Далее 12 переносим перед 3, получаем 1234, что и требовалось.

```
tpolka = array[1..maxbook] of tname;
```

```
var
```

```
  fin,fout : text;
```

```
function fname:string;
```

```
  var
```

```
    s : string;
```

```
  begin
```

```
    write('Enter file name:');
```

```
    readln(s);
```

```
    fname:=s
```

```
  end;
```

```
procedure inp(var polka:tpolka;var num_book:integer);
```

```
var
```

```
  i : integer;
```

```
begin
```

```
  Assign(fin,fname);
```

```
  Reset(fin);
```

```
  readln(fin,num_book);
```

```
  for i:=1 to num_book do
```

```
    readln(fin,polka[i])
```

```
end;
```

```
function findmin(var polka:tpolka;num_book,curnum:integer): integer;
```

```
var
```

```
  temp,i :integer;
```

```
begin
```

```
  if curnum<>0 then
```

```
    temp:=curnum
```

```
  else
```

```
    temp:=curnum+1;
```

```
  for i:=curnum+1 to num_book do
```

```
    if (polka[i]<polka[temp]) then
```

```
      temp:=i;
```

```
  findmin:=temp
```

end;

procedure perest(**var** polka:tpolka;num_book,num1,num2,mesto:**integer**);

var

temp : **array**[1..3] **of** tname;

i : **integer**;

begin

writeln(fout,num1,',',num2,',',mesto);

for i:=num1 **to** num2 **do**

temp[i-num1+1]:=polka[i];

for i:=num2+1 **to** num_book **do**

polka[i-(num2-num1+1)]:=polka[i];

for i:=num_book **downto** mesto+num2-num1+1 **do**

polka[i]:=polka[i-(num2-num1+1)];

for i:=0 **to** num2-num1 **do**

polka[mesto+i]:=temp[i+1];

end;

var

polka : tpolka;

num_book,curnum,min1 : **integer**;

begin

Assign(fout,'output.txt');

Rewrite(fout);

inp(polka,num_book);//считываем из файла начальное положение книг

curnum:=0; //считаем, что изначально у нас 0 упорядоченных книг

while num_book-curnum>2 **do** //будем упорядочивать книги,

//пока не останется 2 книги

begin

min1:=findmin(polka,num_book,curnum+1); //ищем позицию минимальной
//книги, начиная с первой неупорядоченной

if (min1=curnum+1) **then** //если она следующая после упорядоченных, то

inc(curnum)//увеличиваем счётчик упорядоченных книг

//(книга уже стоит на месте)

else //иначе

begin

if min1<num_book **then** //если минимальная книга не последняя, то

perest(polka,num_book,min1,min1+1,curnum+1) //ставим ее вместе со

//следующей на место (после последней упорядоченной)

else //иначе (если минимальная книга последняя)

perest(polka,num_book,min1-2,min1-1,min1-1) //две предпоследние книги

//ставим после последней

end;

end;

if polka[curnum+1]>polka[curnum+2] **then** //если 2 последние книги

//не упорядочены, то

begin

perest(polka,num_book,curnum-1,curnum+1,curnum); //3 предпоследние книги

//переносим в конец

perest(polka,num_book,curnum,curnum+1,curnum-1); //две предпоследних

//переносим на одну позицию в сторону начала

end;

close(fout)

end.