

Какое количество страниц (в тысячах) будет найдено по запросу :

Фрегат | Эсминец?

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

Ответ: _____.

18. На числовой прямой даны два отрезка: $P = [27; 50]$ и $Q = [45; 62]$. Укажите наименьшую возможную длину такого отрезка A , что формула

$$(x \in P) \rightarrow (((x \in Q) \wedge \neg(x \in A)) \rightarrow \neg(x \in P))$$

истинна при любом значении переменной x , то есть принимает значение 1 при любом значении переменной x .

www.ctege.info

Ответ: _____.

19. В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Значения элементов равны 2; 7; 3; 6; 4; 1; 0; 2; 5; 6 соответственно, т.е. $A[0] = 2$; $A[1] = 7$ и т.д.

Определите значение переменной c после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

Бейсик	Паскаль
<pre>c = 0 FOR i = 1 TO 9 IF A(i - 1) > A(i) THEN c = c + 1 t = A(i) A(i) = A(i - 1) A(i - 1) = t ENDIF NEXT i</pre>	<pre>c := 0; for i := 1 to 9 do if A[i - 1] > A[i] then begin c := c + 1; t := A[i]; A[i] := A[i - 1]; A[i - 1] := t end;</pre>

Си	Алгоритмический язык
<pre> c = 0; for (i = 1; i <= 9; i++) if (A[i-1] > A[i]) { c++; t = A[i]; A[i] = A[i - 1]; A[i - 1] = t; } </pre>	<pre> с := 0 нц для i от 1 до 9 если A[i - 1] > A[i] то с := с + 1 t := A[i] A[i] := A[i - 1] A[i - 1] := t все кц </pre>

Ответ: _____.

20. Ниже на четырёх языках записан алгоритм. Получив на вход число x , этот алгоритм печатает два числа: a и b . Укажите такое число x , при вводе которого алгоритм печатает сначала 2, а потом 81.

Бейсик	Паскаль
<pre> DIM X, A, B AS INTEGER INPUT X A=0: B=1 WHILE X > 0 A = A+1 B = B*(X MOD 10) X = X \ 10 WEND PRINT A PRINT B </pre>	<pre> var x, a, b: integer; begin readln(x); a:=0; b:=1; while x>0 do begin a:=a+1; b:=b*(x mod 10); x:= x div 10 end; writeln(a); write(b); end. </pre>
Си	Алгоритмический
<pre> #include<stdio.h> void main() { int x, a, b; scanf("%d", &x); a=0; b=1; while (x>0){ a=a+1; b=b*(x%10); x = x/10; } printf("%d\n%d", a, b); } </pre>	<pre> алг нач цел x, a, b ввод x a:=0; b:=1 нц пока x>0 a:=a+1 b:=b*mod(x,10) x:=div(x,10) кц вывод a, b кон </pre>

Ответ:

21. Напишите в ответе число, которое будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках).

Бейсик	Паскаль
<pre> DIM A, B, T, M, R AS INTEGER A = -11: B = 11 M = A: R = F(A) FOR T = A TO B IF F(T) < R THEN M = T R = F(T) END IF NEXT T PRINT M FUNCTION F(x) F = 2*(x*x-25)*(x*x-25)+5 END FUNCTION </pre>	<pre> var a,b,t,M,R :integer; Function F(x:integer):integer; begin F := 2*(x*x-25)*(x*x-25)+5 end; begin a := -11; b := 11; M := a; R := F(a); for t := a to b do begin if (F(t) < R) then begin M := t; R := F(t) end end; write(M) end. </pre>
Си	Алгоритмический
<pre> #include<stdio.h> int F(int x) { return 2*(x*x-25)*(x*x-25)+5; } void main() { int a, b, t, M, R; a = -11; b = 11; M = a; R = F(a); for (t = a; t <= b; t++) { if (F(t) < R) { M = t; R = F(t); } } printf("%d", M); } </pre>	<pre> алг нач цел a, b, t, M, R a := -11; b := 11 M := a; R := F(a) нц для t от a до b если F(t) < R то M := t; R := F(t) все кц вывод M кон алг цел F(цел x) нач знач:=2*(x*x-25)*(x*x-25)+5 кон </pre>

Ответ: _____

22. У исполнителя Утроитель две команды, которым присвоены номера:

1. прибавь 1,

2. умножь на 3.

Первая из них увеличивает число на экране на 1, вторая утраивает его.

Программа для Утроителя — это последовательность команд.

Сколько есть программ, которые число 2 преобразуют в число 14?

Ответ: _____.

23. Каково наибольшее целое число x , при котором ложно высказывание

$$(8 \cdot x - 6 < 75) \rightarrow (x \cdot (x - 1) > 65)?$$

Ответ: www.ctege.info _____.

Не забудьте перенести все ответы в бланк ответов № 1.

Часть 2

Для записи ответов к заданиям этой части (24–27) используйте бланк ответов № 2. Запишите сначала номер задания (24, 25 и т.д.), а затем полное решение. Ответы записывайте чётко и разборчиво.

24. Требовалось написать программу, которая решает неравенство $(x - a)/(bx) > 0$ относительно x для любых ненулевых чисел a и b ($b \neq 0$, $a \neq 0$), введённых с клавиатуры. Все числа считаются действительными. Программист торопился и написал программу неправильно.

ПРОГРАММА НА ПАСКАЛЕ	<pre>var a,b,x: real; begin readln(a,b,x); if b>0 then write ('x>',a,' или x <0') else if a > 0 then write('0 < x <', a) else write(a,'< x <0'); end.</pre>
ПРОГРАММА НА БЕЙСИКЕ	<pre>INPUT a, b, x IF b > 0 THEN PRINT "x>",a," или x<0" ELSE IF a > 0 THEN PRINT "0<x<",a ELSE PRINT a,"<x<0" ENDIF ENDIF END</pre>
ПРОГРАММА НА СИ	<pre>void main(void) { float a,b,x; scanf("%f%f%f", &a,&b,&x); if (b>0) printf("x>%f или x<0 ", a); else if (a>0) printf("0<x<%f",a); else printf("%f<x<0",a); }</pre>

Последовательно выполните три задания:

- 1) Приведите пример таких чисел a , b , x , при которых программа неверно решает поставленную задачу.
- 2) Укажите, какая часть программы является лишней.
- 3) Укажите, как нужно доработать программу, чтобы не было случаев её неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы.)

25. Опишите на русском языке или одном из языков программирования алгоритм подсчёта максимального количества подряд идущих элементов, каждый из которых больше предыдущего, в целочисленном массиве длины 30.

26. Два игрока играют в следующую игру. Перед ними лежат две кучки камней, в первой из которых 1, а во второй — 2 камня. У каждого игрока неограниченно много камней. Игроки ходят по очереди. Ход состоит в том, что игрок или увеличивает в 3 раза число камней в какой-то куче, или добавляет 4 камня в какую-то кучу.

Выигрывает игрок, после хода которого в одной из куч становится не менее 20 камней. Кто выигрывает при безошибочной игре обоих игроков — игрок, делающий первый ход, или игрок, делающий второй ход? Каким должен быть первый ход выигрывающего игрока? Ответ обоснуйте.

27. На вход программе подаются сведения об учениках некоторой средней школы. В первой строке сообщается количество учеников N , каждая из следующих N строк имеет следующий формат: <Фамилия> <Имя> <класс>, где <Фамилия> — строка, состоящая не более чем из 20 символов, <Имя> — строка, состоящая не более чем из 15 символов, <класс> — год обучения (от 1 до 12) и заглавная буква (от «А» до «Я») без пробела. <Фамилия> и <Имя>, а также <Имя> <класс> разделены одним пробелом. Пример входной строки:

Иванов Петр 10Б

Требуется написать программу на языке Паскаль или Бейсик, которая будет выводить на экран ин-

формацию о параллелях (годе обучения) с наибольшим числом учеников. Программа должна выводить на экран в первой строке количество учеников в искомых параллелях, а во второй строке — в порядке возрастания номера этих параллелей через пробел. Например:

100

1 7 11

www.ctege.info

Вариант 8

24. 1) $a = -1, b = 1, x = 0$.

Значение x может быть не указано. Значение a может быть любым отрицательным числом, значение b — любым положительным. Также допустим ответ, что программа работает неправильно при любых положительных b и отрицательных a .

2) Лишняя часть:

не нужно вводить x с клавиатуры;

верно: `readln(a, b)`.

3) Возможная доработка:

```
readln(a, b);
if b > 0 then
if a > 0 then
write('x > ', a, ' или x < 0')
else
write('x < ', a, ' или x > 0')
else
if a > 0 then
write('0 < x < ', a)
else
write(a, ' < x < 0');
```

(могут быть и другие способы доработки).

25. *Пример правильного описания алгоритма на русском языке.*

Заводим переменную `MaxIncr` для хранения максимального количества подряд идущих в порядке возрастания элементов и счётчик `NumIncr` для хранения числа элементов в последней группе возрастающих элементов. Просматривая элементы массива, сравниваем очередной элемент со следующим за ним. Если очередной элемент массива оказывается не меньше следующего, то сравниваем текущее значение счётчика со значением переменной `MaxIncr`; если он больше, то заменяем значение переменной `MaxIncr` значением

счётчика, при этом значение NumIncr обнуляется. Так повторяем до конца массива. В конце работы нужно ещё раз сравнить значение счётчика со значением переменной MaxIncr и переопределить её, если счётчик больше.

Пример правильной и эффективной программы (на основе алгоритма, использующего однократный проход по массиву):

На языке Паскаль

```
const N = 30;
var a:array[1..N] of integer;
MaxIncr, NumIncr, i: integer;
begin
  MaxIncr:=0;
  NumIncr:=0;
  for i:=1 to N-1 do
  begin
    if a[i]<a[i+1] then
      NumIncr:=NumIncr+1;
    else begin
      if NumIncr>MaxIncr then
        MaxIncr:=NumIncr;
      NumIncr:=0;
    end;
  end;
  if NumIncr>MaxIncr then
    MaxIncr:=NumIncr;
  writeln(MaxIncr);
end.
```

На языке Бейсик

```
N = 30
DIM i, MaxIncr, NumIncr, a(N) AS INTEGER
MaxIncr=0
NumIncr=0
FOR i = 1 TO N-1
IF a(i)<a(i+1) THEN
  NumIncr=NumIncr+1
ELSE
  IF NumIncr>MaxIncr THEN
    MaxIncr=NumIncr
  ENDIF
  NumIncr=0
ENDIF
NEXT i
```

```

IF NumIncr>MaxIncr THEN
  MaxIncr=NumIncr
ENDIF
PRINT MaxIncr
END

```

26. Выигрывает второй игрок.

Для доказательства рассмотрим неполное дерево игры, оформленное в виде таблицы, где в каждой ячейке записаны пары чисел, разделённые запятой. Эти числа соответствуют количеству камней на каждом этапе игры в первой и второй кучах соответственно.

	1-й ход	2-й ход	3-й ход	4-й ход	
Стартовая позиция	I-й игрок (все варианты хода)	II-й игрок (выигрышный ход)	I-й игрок (все варианты хода)	II-й игрок (один из вариантов)	Пояснение
1,2	3,2	3,6	9,6	27,6	Второй игрок выигрывает на четвёртом ходу, после любого ответа первого игрока, например, утроив число камней в самой большой куче
			7,6	21,6	
			3,18	3,54	
			3,10	3,30	
	5,2	5,6	15,6	45,6	Второй игрок выигрывает на четвёртом ходу после любого ответа первого игрока, например, утроив число камней в самой большой куче
			9,6	27,6	
			5,18	5,54	
			5,10	5,30	
	1,6	3,6 или 5,6	Те же варианты третьего-четвёртого ходов		

Таблица содержит *все возможные* варианты ходов первого игрока. Из неё видно, что при любом ходе первого игрока у второго имеется ход, приводящий к победе.

27. Программа должна верно читать входные данные, не запоминая их все, а сразу подсчитывая в массиве, хранящем 12 целых чисел, количество учащихся в каждой из параллелей. Затем с использованием этого массива ищется параллель с максимальным числом учеников. За дополнительный просмотр этого массива распечатывается информация об искомых параллелях. Баллы начисляются только за программу, которая решает задачу хотя бы для частного случая (например, параллель с максимальным количеством учеников единственна).

Пример правильной и эффективной программы на языке Паскаль:

```
var pc:array[1..12] of integer;
    p:1..12;
    class:string[3];
    c:char;
    max, i, N:integer;
begin
    readln(N);
    for i:=1 to 12 do
        pc[i]:=0;
    for i:=1 to N do
        begin
            repeat
                read(c)
            until c=' '; {считана фамилия}
            repeat
                read(c)
            until c=' '; {считано имя}
            readln(class);
```

```

    {определяем номер параллели}
    if length(class)=2 then
    p:=ord(class[1])-ord('0') else
    p:=(ord(class[1])-ord('0'))*10+
    ord(class[2])-ord('0');
    pc[p]:=pc[p]+1;{учитываем ученика этой
параллели}
end;
max:=0;
for i:=1 to 12 do
    if pc[i]>max then max:=pc[i];
writeln('Максимум учеников в параллели:',max);
for i:=1 to 12 do
    if pc[i]=max then
        write(i, ' ');
readln
end.

```

Пример правильной программы на языке Бейсик:

```

DIM i, j, p, n, max, pc(12) AS INTEGER
DIM m(12)
DIM s AS STRING
FOR i = 1 TO 12
    pc(i) = 0
NEXT i
INPUT n
FOR j = 1 TO n
LINE INPUT s
c$ = MID$(s, 1, 1)
i = 1
WHILE NOT (c$ = " ")
    i = i + 1
    c$ = MID$(s, i, 1)
WEND
i = i + 1
c$ = MID$(s, i, 1)
WHILE NOT (c$ = " ")
    i = i + 1
    c$ = MID$(s, i, 1)

```

```

WEND
s = MID$(s, i + 1, 3)
IF MID$(s, 2, 1) >= "0" AND MID$(s, 2, 1) <= "2"
THEN
    p=(ASC(MID$(s,1,1))-ASC("0"))*10+ASC(MID$(s,2,1))-ASC("0")
ELSE
    p = ASC(MID$(s, 1, 1)) - ASC("0")
ENDIF
pc(p) = pc(p) + 1
NEXT j
max = 0
FOR i = 1 TO 12
    IF pc(i) > max THEN max = pc(i)
NEXT i
PRINT "Max = "; max
FOR i = 1 TO 12
    IF pc(i) = max THEN PRINT i; " ";
NEXT i
END

```

www.ctege.info

Задания	1	2	3	4	5	6	7	8	9	10	11	
Вариант 8	1	3	4	1	24	1222	2	10	140000	2	30	
Задания	12	13	14	15	16	17	18	19	20	21	22	23
Вариант 8	DCEA	180	2	4	1020	6200	5	8	99	-5	4	8