



Вариант № 1

№9

// Pascal ~~(TP)~~ (Turbo Pascal x16) (КОИ-8R / Cp866)

12

Const
Prefabs: Array[0..9] of String[6] = (
 'ОДИН--', 'ДВА---', 'ТРИ---', 'ЧЕТЫРЕ', 'ПЯТЬ--',
 'ШЕСТЬ--', 'СЕМЬ--', 'ВОСЕМЬ', 'ДЕВЯТЬ', 'НОЛЬ--'
); {Из-за различной длины нужных слов приходится вводить символ-ограничитель.}

Var
Str: String[16];
LastPos: Byte; {Большее 255 строка в 16 символов обеспечить не сможет}

Procedure Init();
Begin
 Read(Str);
 ~~Start~~ LastPos := 0;
End;

Procedure CheckForSubstring(Substr: String[6]);

Var
i, j: Byte;
ErrorFlag: Boolean;

Begin
 While Ord(Str[0]) > 0 do {В Turbo Pascal длина строки задаётся кодом }
 Begin {ASCII-символа на месте "0". Обраща- }
 If Str[1] = Substr[1] then {ться напрямую к памяти быстрее, чем }
 Begin {получать значение "Length(Str)"}
 ErrorFlag := False;
 i := 2;
 While not ErrorFlag and (Substr[i] <> '-') and (i <= Ord(Substr[0]))
 Begin
 If (Str[i] <> Substr[i]) or (i > Ord(Str[0]))
 ErrorFlag := True;
 Else
 ~~Begin~~
 ~~Inc(i)~~ Inc(i);
 End;
 If not ErrorFlag then
 Begin
 LastPos := i;
 Break; {Здесь делает то же, что и "Exit()": сбрасывает все циклы в }
 {Здесь, => выходит из процедуры.}
 End;
 End;
 End;
 Delete(Str, 1, 1);
 WriteLn('Удалено: ', Str[1]);
 End;
 End;
End; {of Procedure}

(continue) →



Nº 9 (cont.)

{ Сама программа }

Begin

Int(),

For $i = 0$ to 9 do

Begin

```
Begin  
CheckForSubstring ( Prefab[i] );
```

If StartPos $\neq$ 0 then

Break;

End,

For $i = \text{LastPos} + 1$ to $\text{Ord}(\text{Str}[0])$ do

Begin

```
WriteLn ( 'УКАЖЕМО: ', Str [ LastPos + 1 ] );
```

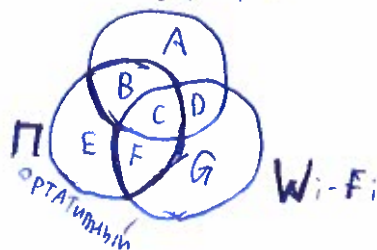
Delete (Str, 1, 1)

End,

WriteLn(Str); {ОСТАВШАЯСЯ СТРОКА}

End.

№ 3 (4) 5
Микроскоп



$ABCD = 50$
 $BCFE = 60$
 $CDFG = 70$
 $ABCDEF = 80$
 $ABCDGF = 100$
 $CDF = 20$
 $BCF = 7$

$$\begin{aligned} G &= CDFG - CDF = 50, \\ B_{ce}(V) &= ABCDEF + G = 130, \\ E &= V - ABCDFG = 30, \\ BCF &= BCEF - E = 30. \end{aligned}$$

Amber: 30

205

$$4095_d = 4096 - 1 = 333333_4$$

$\frac{123}{3}$ $\frac{101112132021222330313233}{3 \cdot 4 = 12}$ $\frac{100101\dots}{3 \cdot 16 = 48}$

Всего цифр: $3 \cdot \underbrace{1}_{4^0} + 3 \cdot \underbrace{4}_{4^1} + 3 \cdot \underbrace{16}_{4^2} + 3 \cdot \underbrace{64}_{4^3} + 3 \cdot \underbrace{256}_{4^4} + 3 \cdot \underbrace{1024}_{4^5} = 3(1 + 4 + 16 + 64 + 256 + 1024) = 4095$ цифр в сист. счисления с осн. 4.

binary: $\underbrace{10111}_{\text{base 4}} \underbrace{10001}_{\text{base 4}}$
 $\underbrace{\hspace{1.5cm}}_{\text{base 16}} \quad \underbrace{\hspace{1.5cm}}_{\text{base 16}}$

Чем в 4-значной (с округл. вверх) $\Rightarrow B \text{ base } 16: \left\lceil \frac{4095}{2} \right\rceil = 2048$.

Ombem: 2048.



Вариант № 1

№ 6

9

```
int F(int N) {  
    if (N % 5 == 2) {  
        return F(N/5 - 2) + F(N/5);  
        //  $\log_{10} 10^N = N \cdot \log_{10} 10 = N \cdot 1 = N$ , т.е. логиты и не смотрят  
        // на это число -  
        // еще.  
    } else {  
        if (N <= 0) {  
            return 1;  
            //  $\ln(N^N) / \ln N = N \log(N) = N \cdot 1 = N$ , т.е. также.  
        } else {  
            return F(N - 4) + F(N - 5);  
        }  
    }  
} // of int F(int N) {
```

$F(32): F(4) + F(6); = 2 + 4 = 6$
 $F(6): F(2) + F(1); = 2 + 2 = 4$
 $F(2): F(-2) + F(0); = 1 + 1 = 2$
 $F(1): F(-3) + F(-4); = 1 + 1 = 2$
 $F(4): F(0) + F(-1); = 1 + 1 = 2$
 $F(0): 1 = 1$
 $F(-1...-\infty): 1 = 1$

Ответ: 6.

№ 10

// Borland C

```
#include <stdio.h>  
unsigned int x;  
unsigned int a[10000], short int flag;  
void main(void) {  
    scanf("%i", &x); flag = 0;  
    for(int i = 0; i < 10000; i++) {  
        a[i] = 1; // (простое число. Находится только передо мной всех значений  
        // до из него, алгоритмов других нет (они не точны))  
        // (простые, все умеваю. Знаю, как, по нецен.)  
        if (a[i] == x) { flag = 1; break; }  
    }  
}
```

15



№ 18 (продолж.)

Олимпиада школьников «Гранит науки»

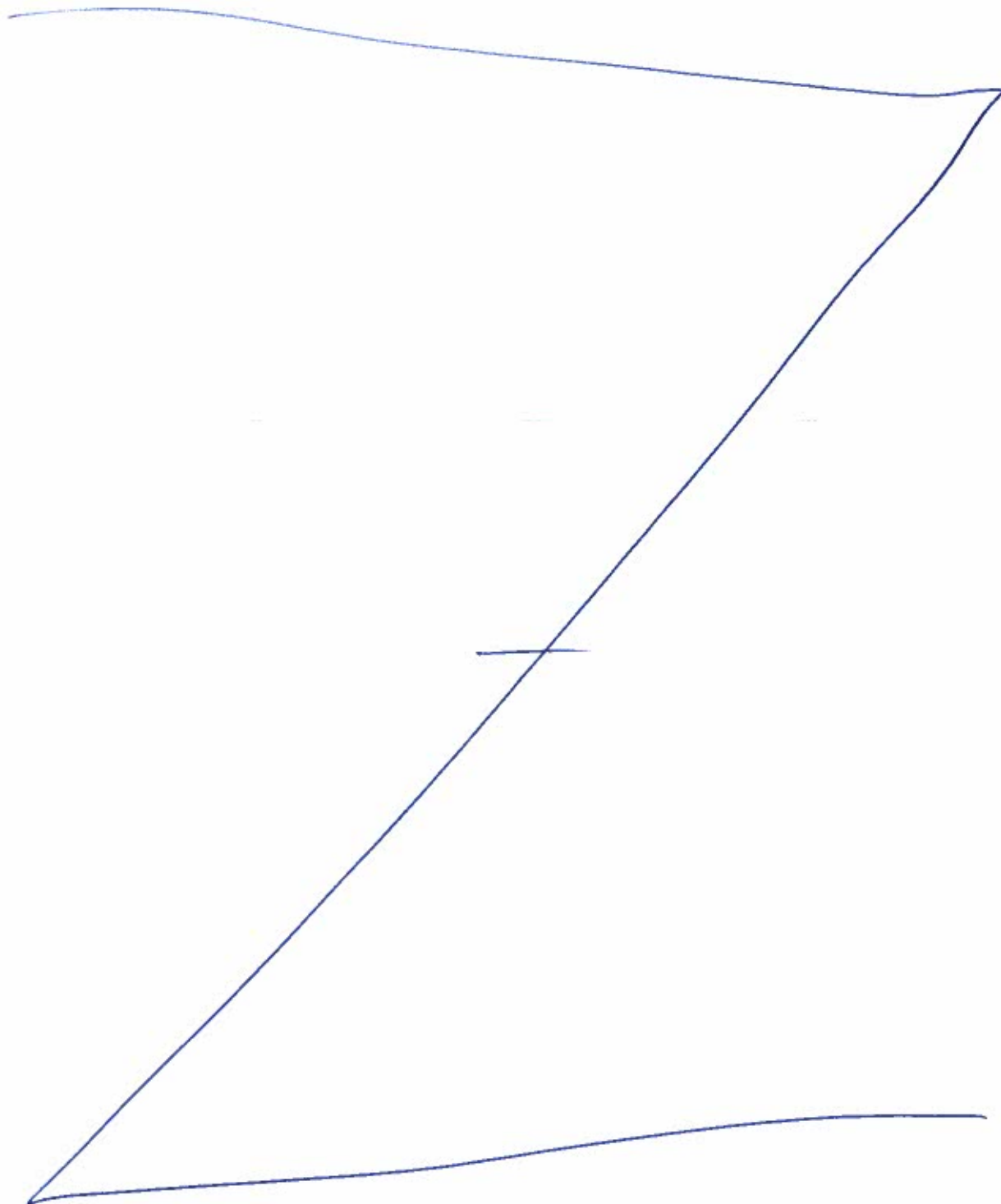
БЛАНК ОЛИМПИАДНОЙ РАБОТЫ

```
if (flag != 1) {
    for (int i = 0; i < a.sizeof(), i++) {
        for (int j = 0; j < x; j++) {
            if (a[j] + a[i] == x) {
                flag = 2;
                break;
            }
            if (a[j] > a[i]) j = x;
        }
    }
}

if (flag == 2)
    printf("%i явл. суммой %i и %i.", x, a[j], a[i]);
else
    printf("%i не явл. суммой прост. чисел.", x);
} // of void main (void)
```



Вариант № 1





Олимпиада школьников «Гранит науки»
БЛАНК ОЛИМПИАДНОЙ РАБОТЫ

