Języki formalne i techniki translacji
Laboratorium - Projekt - Testy

program0.imp

```
VAR
    a b
BEGIN
    READ a;
    WHILE a > 0 DO
        b := a / 2;
        b := 2 * b;
        IF a > b THEN WRITE 1;
        ELSE WRITE 0;
        ENDIF
        a := a / 2;
    ENDWHILE
END
```

program1.imp

```
( sito Eratostenesa )
VAR
    n j sito[100]
BEGIN
    n := 100-1;
    FOR i FROM n DOWNTO 2 DO
        sito[i] := 1;
    ENDFOR
    FOR i FROM 2 TO n DO
        IF sito[i] <> 0 THEN
            j := i + i;
            WHILE j <= n DO
                sito[j] := 0;
                j := j + i;
            ENDWHILE
            WRITE i;
        ENDIF
    ENDFOR
END
```

program2.imp

```
( Rozklad liczby na czynniki pierwsze )
VAR
    n m reszta potega dzielnik
BEGIN
    READ n;
    dzielnik := 2;
    m := dzielnik * dzielnik;
    WHILE n >= m DO
        potega := 0;
        reszta := n % dzielnik;
        WHILE reszta = 0 DO
```

```
            n := n / dzielnik;
            potega := potega + 1;
            reszta := n % dzielnik;
        ENDWHILE
        IF potega > 0 THEN ( czy znaleziono dzielnik )
            WRITE dzielnik;
            WRITE potega;
        ELSE
            dzielnik := dzielnik + 1;
            m := dzielnik * dzielnik;
        ENDIF
    ENDWHILE
    IF n <> 1 THEN ( ostatni dzielnik )
        WRITE n;
        WRITE 1;
    ENDIF
END
```

0-div-mod.imp

```
( div-mod.imp
  1 0
  1 0 0 0
)
VAR
        a b c
BEGIN
        READ a;
        READ b;
        c := a / a;
        WRITE c;
        c := a / b;
        WRITE c;
        c := a % a;
        WRITE c;
        c := a % b;
        WRITE c;
END
```

1-numbers.imp

```
( numbers.imp - liczby )
VAR
        a b c t[7] d e f g h i j tab[6]
BEGIN
        WRITE 0;
        WRITE 1;
        WRITE 2;
        WRITE 10;
        WRITE 100;
        WRITE 10000;
        WRITE 1234567890;

        a := 1234566543;
        b := 677777177;
        c := 15;
        t[2] := 555555555;
        d := 8888;
        tab[4] := 11;
        t[0] := 999;
        e := 1111111111;
        tab[0] := 7777;
        f := 2048;
        g := 123;
        t[3] := t[0];
        tab[5] := a;
        tab[5] := tab[0] / tab[4];
        t[5] := tab[0];

        READ h;
        i := 1;
        j := h + c;

        WRITE j; ( j = h + 15 )
        WRITE c; ( c = 15 )
        WRITE t[3]; ( 999 )
        WRITE t[2]; ( 555555555 )
        WRITE t[5]; ( 7777 )
        WRITE t[0]; ( 999 )
        WRITE tab[4]; ( 11 )
        WRITE tab[5]; ( 707 )
        WRITE tab[0]; ( 7777 )
END
```

2-fib.imp

```
( Fibonacci 26
? 1
> 121393
)
VAR
  tab[1234567890] a b c d e f g h i j k l m n o p q r s t u v w x y z
BEGIN
  READ tab[12345];
  a:=tab[12345];
  b:=a;
  c:=b+a;
  d:=c+b;
  e:=d+c;
  f:=e+d;
  g:=f+e;
  h:=g+f;
  i:=h+g;
  j:=i+h;
  k:=j+i;
  l:=k+j;
  m:=l+k;
  n:=m+l;
  o:=n+m;
  p:=o+n;
  q:=p+o;
  r:=q+p;
  s:=r+q;
  t:=s+r;
  u:=t+s;
  v:=u+t;
  w:=v+u;
  x:=w+v;
  y:=x+w;
  z:=y+x;
  tab[a]:=z;
  WRITE tab[a];
END
```

3-fib-factorial.imp

```
( Silnia + Fibonacci
? 20
> 2432902008176640000
> 17711
)
VAR
    f[101] s[101] i[101] n k l
BEGIN
    READ n;
    f[0]:=1;
    s[0]:=1;
    i[0]:=0;
    FOR j FROM 1 TO n DO
                k:=j-1;
        l:=k-1;
                i[j]:=i[k]+1;
                f[j]:=f[k]+f[l];
        s[j]:=s[k]*i[j];
    ENDFOR
    WRITE s[n];
    WRITE f[n];
END
```

4-factorial.imp

```
( Silnia
? 20
> 2432902008176640000
)
VAR
  s[101] n m a j
BEGIN
    READ n;
    s[0]:=1;
    m:=n;
    FOR i FROM 1 TO m DO
                a:=i%2;
                j:=i-1;
                IF a=1 THEN
                        s[i]:=s[j]*m;
                ELSE
                        s[i]:=m*s[j];
                ENDIF
                m:=m-1;
    ENDFOR
    WRITE s[n];
END
```

5-tab.imp

```
( tab.imp )
VAR
        n j ta[25] tb[25] tc[25]
BEGIN
        n := 25 - 1;
        tc[0] := n;
        tc[n] := n - n;
        FOR i FROM tc[0] DOWNTO tc[n] DO
                ta[i] := i;
                tb[i] := n - i;
        ENDFOR
        FOR i FROM tc[n] TO tc[0] DO
                tc[i] := ta[i] * tb[i];
        ENDFOR
        FOR i FROM 0 TO n DO
                WRITE tc[i];
        ENDFOR
END
```

6-mod-mult.imp

```
( a ^ b mod c
? 1234567890
? 1234567890987654321
? 987654321
> 674106858
)
VAR
    a b c wynik pot wybor
BEGIN
    READ a;
    READ b;
    READ c;
    wynik:=1;
    pot:=a%c;
    WHILE b>0 DO
                wybor:=b%2;
                IF wybor=1 THEN
                        wynik:=wynik*pot;
                        wynik:=wynik%c;
                ENDIF
                b:=b/2;
                pot:=pot*pot;
                pot:=pot%c;
    ENDWHILE
    WRITE wynik;
END
```

7-loopiii.imp

```
( loopiii.imp - zagniezdzone petle
        0 0 0
        31000 40900 2222010

        1 0 2
        31001 40900 2222012
)
VAR
        a b c
BEGIN
        READ a;
        READ b;
        READ c;
        FOR i FROM 111091 TO 111110 DO
                FOR j FROM 209 DOWNTO 200 DO
                        FOR k FROM 11 TO 20 DO
                                a := a + k;
                        ENDFOR
                        b := b + j;
                ENDFOR
                c := c + i;
        ENDFOR
        WRITE a;
        WRITE b;
        WRITE c;
END
```

8-for.imp

```
( for.imp
  12 23 34
  507 4379 0
)
VAR
        a b c
BEGIN
        READ a;
        READ b;
        READ c;
        FOR i FROM 9 DOWNTO 0 DO
                FOR j FROM 0 TO i DO
                        FOR k FROM 0 TO j DO
                                a := a + k;
                                c := k * j;
                                c := c + i;
                                b := b + c;
                        ENDFOR
                ENDFOR
        ENDFOR
        WRITE a;
        WRITE b;
        WRITE c;
END
```

9-sort.imp

```
( sort.imp
)
VAR
        tab[22] x q w j k n m
BEGIN
        n := 23;
        m := n - 2;
        q := 5;
        w := 1;
        (generowanie nieposortowanej tablicy)
        FOR i FROM 0 TO m DO
                w := w * q;
                w := w % n;
                tab[i] := w;
        ENDFOR
        (wypisywanie nieposortowanej tablicy)
        FOR i FROM 0 TO m DO
                WRITE tab[i];
        ENDFOR
        WRITE 1234567890;
        (sortowanie)
        FOR i FROM 1 TO m DO
                x := tab[i];
                j := i;
                WHILE j > 0 DO
                        k := j - 1;
                        IF tab[k] > x THEN
                                tab[j] := tab[k];
                                j := j - 1;
                        ELSE
                                k := j;
                                j := 0;
                        ENDIF
                ENDWHILE
                tab[k] := x;
        ENDFOR
        (wypisywanie posortowanej tablicy)
        FOR i FROM 0 TO m DO
                WRITE tab[i];
        ENDFOR
END
```