

Dane są dwie liczby typu float pojedynczej precyzji a i b.  
Wyznaczyć wartość bezwzględną iloczynu tych liczb:

```
(float a=-10; float b=10; float wynik=0;)
    FLD word ptr a;
    FLD word ptr b;
    FMUL;
    FABS;
    FSTP word ptr wynik;
```

---

#### **SORTOWANIE BĄBELKOWE:**

```
int N; // rozmiar tablicy
float* A=new float[n];
for(int i=0;i<n;i++)
{ cin>>A[i]; }
__asm{
    MOV ecx, N;
    MOV esi, A;
petla2:
    PUSH ecx;
    MOV ecx, N;
    FLD [esi + 4 * ecx - 4 ];
    DEC ecx;
petla1:
    FLD [esi + 4 * ecx - 4];
    FCOM st(1);
    FSTSW ax;
    SAHF;
    JC pomin;
    FXCH;
pomin:
    FSTP [esi + 4 * ecx];
    DEC ecx;
    JNZ petla1;
    FSTP [esi];
    POP ecx;
    DEC ecx;
    JNZ petla2;
}
```

---

Dane są dwie liczby typu float a i b.  
Zbadać ile razy b mieści się w a.

```
    FLD a;
    FLD b;
    FDIV;
    FSTP wynik;
```

---

Dana jest zmienna alfa typu float reprezentująca  
wartość kąta wyrażoną w stopniach. Obliczyć  
wartość:

```
wart = tan(alfa) + 1 / c*tan(alfa)
float wart=0;
    FLD alfa;
    FPTAN;
    FLD st;
    FMUL c;
    FLD1;
    FDIV;
    FADD;
    FSTP wart;
```

---

Dana jest jednowymiarowa tablica dynamiczna wek  
typu double o liczbie elementów n reprezentująca  
wektor.

Podnieść wszystkie elementy tego wektora do  
kwadratu i umieścić je w tej samej tablicy wek.

```
    MOV ecx, n; // rozmiar
    MOV edi, tab; // adres początku wektora
    FLD qword ptr[edi]; // [tab[0] na wierzcholek
    FLD st; // to samo
    FMULP st(1), st;
    FST qword ptr [edi]; // sciagniecie ze stosu
    DEC ecx;
petla:
    ADD edi, 8; // nastepny element wektora
    FLD qword ptr[edi]; //tab[1] na wierzcholek
    FLD st;
    FMULP st(1), st;
    FSTP qword ptr[edi];
    LOOP petla;
```

---

Dla zmiennej x typu double obliczyć  $y = e^x$

```
    FLD x;           x
    FLDE;            e; x
    FYL2X;           x*log2(e)
    FLD st;          x*log2(e); x*log2(e)
    FRNDINT;
    FSUB st(1), st;
    FXCH;
    F2XM1;
    FLD1;
    FADD;
    FSCALE;
    FSTP y;
```

---

## TRYGONOMETRIA SINUS / COSINUS

Alfa wprowadzana z klawiatury

float alfa, float wynik, float zmienna180;

```
FLD alfa;
FLDPI;
FMUL;
FLD zmienna180;
FDIV;
FSIN; // jak cos to FCOS
FSTP wynik;
```

---

Oblicz  $y = \sqrt{|a-b|} / (a+b)$

```
FLD a;          // a
FLD st(0);       // a; a
FLD b;          // b; a; a
FADD st(2), st;  // b; a; a+b
FSUBP st(1), st; // a-b; a+b
FABS;           // |a-b|; a+b
FSQRT;          // sqrt(|a-b|); a+b
FDIVR;          // sqrt(|a-b|)/a+b
FSTP y;
```

---

Logarytm o dowolnej podstawie i dowolnej liczbie logarytmowanej.

float wynik, x, b (x – logarytmowana, b – podstawa)

```
FLD1;          // 1
FLD x;          // x, 1
FYL2X;         // log_2 x
FLD1;          // 1, log_2 x
FLD b;          // b, 1, log_2 x
FYL2X;         // log_2 b, log_2 x
FDIV;          // WYNIK
FSTP wynik;
```

---

Obliczenie przeciwprostokątnej  $a^2+b^2 = c^2$

```
FLD a;
FLD st(0);
FMUL;
FLD b;
FLD st(0);
FMUL;
FADD;
FSQRT;
FSTP c;
```

---

Wyznaczyć minimum z dwóch liczb:

```
FLD x;
FLD y;
FCOMI st(0), st(1);
JC etykieta;
FXCH;
```

etykieta:

```
FSTP wynik;
```

---

Wyznaczyć maksimum z dwóch liczb:

```
FLD x;
FLD y;
FCOMI st(0), st(1);
JNC etykieta;
FXCH;
```

Etykieta:

```
FSTP wynik;
```

---

$Y = \min(1 - a, 1 - b)$

```
FLD1;
FLD b;
FSUB;
FLD1;
FLD a;
FSUB;
FCOMI st, st(1);
JC koniec; // jak maksimum to JNC
FXCH;
```

koniec:

```
FSTP y;
```

---

$y = a * \sin(x) + \cos(x)/b$

float zmienna180 = 180.0;

```
FLD x;
FLDPI;
FMUL;
FLD zmienna180;
FDIV;
FSIN;
FLD a;
FMUL;
FLD x;
FLDPI;
FMUL;
FLD zmienna180;
FDIV;
FCOS;
FLD b;
FDIV;
FADD;
FSTP wynik;
```

---

---

$$Y = a - \cos(x) / b - \sin(x)$$

```
FLD a;
FLD x;
FCOS;
FSUB;
FLD b;
FLD x;
FSIN;
FSUB;
FDIV;
FSTP y;
```

---

Wektor A 86-elementowy wartości  
zmiennoprzecinkowych pojedynczej precyzji,  
zliczyć sumę wartości absolutnych.

```
PUSH esi;
MOV esi, A;
MOV ecx, rozmiar; // tutaj jest 86
FLDZ;
```

Petla1:

```
FLD dword ptr [esi + 4 * ecx - 4];
FABS;
FADD;
DEC ecx;
JNZ Petla1;
FSTP wynik;
```

---

Oblicz pole powierzchni całkowitej p ostrosłupa  
prawidłowego czworokątnego o boku podstawy a i  
wysokości ściany bocznej h.  
(wzór matematyczny po skróceniu:  $2 * a * h + a^2$ )

```
FLD a;
FLD st(0);
FMUL;
FLD h;
FLD a;
FMUL;
FLD1;
FLD st(0);
FADD;
FMUL;
FADD;
FSTP wynik;
```

Objętość prostopadłościanu:

```
FLD a;
FLD h;
FMUL;
FLD b;
FMUL;
FSTP objetosc;
```

---

Pole boczne prostopadłościanu

```
Float poleboczne, float dwa = 2;
FLD a;
FLD h;
FMUL st(1), st(0);
FLD w;
FMUL st(2), st(0);
FLD b;
FMUL st(2), st(0);
FMUL;
FADD;
FSTP poleboczne;
```

---

Pole powierzchni całkowitej prostopadłościanu o  
bokach rzeczywistych a, b i c:

Zmienna float dwa = 2;

```
FLD a;
FLD b;
FMUL;
FMUL dwa;
FLD a;
FLD c;
FMUL;
FMUL dwa;
FLD b;
FLD c;
FMUL;
FMUL dwa;
FADD;
FADD;
FSTP wynik;
```

---

$$Y = \arctan ( \sin(x) - \cos(x) / \sqrt{a^2 + 1} )$$

```
FLD x;
FSIN;
FLD y;
FCOS;
FSUB;
FLD1;
FLD a;
FLD st(0);
FMUL;
FADD;
FSQRT;
FPATAN;
```

## Iloczyn dwóch wektorów:

```
double *v1, double *v2, int n
    MOV ecx, n;
    MOV esi, v1;
    MOV edi, v2;
    FLD qword ptr[esi]; // v1[0]
    FLD qword ptr[edi]; // v2[0], v1[0]
    FMUL; // s = v2[0] * v1[0]
    DEC ecx;
Petla:
    ADD esi, 8; // bo double – 8 bajtów
    ADD edi, 8;
    FLD qword ptr[esi];
    FMUL qword ptr[edi];
    FADD; // s += v2[i] * v1[i]
    DEC ecx;
    JNZ Petla;
    FSTP iloczyn;
```

---

## Wyrażenie $ax^3 + bx^2 + cx + d$ / float wynik;

```
FLD d;
FLD x;
FLD st(0);
FMUL st(0), st(1);
FLD st(1);
FMUL st(0), st(1);
FMUL a;
FADDP st(3), st(0);
FMUL b;
FADDP st(2), st(0);
FMUL c;
FADD;
FSTP wynik;
```

---

## Wyrażenie $ax^3 + c(b-a) + a(c-b)$

```
FLD x;
FLD st(0);
FMUL st(0), st(0);
FMUL;
FLD a;
FMUL st(1), st(0);
FLD b;
FLD st(0);
FSUB st(0), st(2);
FLD c;
FMUL st(1), st(0);
FSUBRP st(2), st(0);
FADDP st(3), st(0);
FMUL;
FADD;
FSTP wynik;
```

$$Y = | -\arctan(x) + 3/\pi |$$

```
Float z = 3.0;
    FLD x;
    FLD1;
    FPATAN;
    FCHS;
    FLD z;
    FLDPI;
    FDIVP st(1), st(0);
    FADDP st(1), st(0);
    FABS;
    FSTP y;
```

---

$$Y = -(\operatorname{ctg}(|x|))$$

```
FLD x;
FABS;
FLD st(0);
FCOS;
FXCH;
FSIN;
FDIVP st(1), st(0);
FCHS;
FSTP y;
```

---

$$Y = \ln(\sqrt{x^2 + \pi})$$

```
FLD1;
FLDPI;
FLD x;
FMUL x;
FADDP st(1), st(0);
FSQRT;
FYL2X;
FLDL2E;
FDIV;
FSTP y;
```

---

$$Y = -| \operatorname{tg}(x) + 1 |$$

```
FLD x;
FPTAN;
FADD;
FABS;
FCHS;
FSTP y;
```

---

Wypisanie tablicy ciągiem geometrycznym o pierwszym wyrazie a1 i iloczynie q:

```
MOV eax, 0;
MOV edi, tab;
MOV ecx, N-1;
FLD q;
FLD a1;
Petla:
FST qword ptr [edi + 8 * eax];
FMUL st(0), st(1);
INC eax;
LOOP Petla;
FSTP qword ptr [edi + 8 * eax];
FSTP tmp;
```

---

Program obliczający:

If x > b then y:= 2x  
Else y:= 3x

```
FLD x;
FLD b;
FCOMIP st(0), st(1);
JB pomin;
FLD x;
FADDP st(1), st(0);
pomin:
FLD x;
FADDP st(1), st(0);
FSTP y;
```

---

$Y = \sin(a * x) + \text{ctg}(x / b)$ , kąt podany w stopniach

```
PUSH 180;
FLD a;
FLD x;
FMULP st(1), st(0);
FSIN;
FLDPI; // zamiana radianów na stopnie
FMULP st(1), st(0);
FILD dword ptr [esp];
FDIVP st(1), st(0);
FLD x;
FLD b;
FDIVP st(1), st(0);
FPTAN;
FDIVP st(1), st(0);
FLDPI;
FMULP st(1), st(0);
FILD dword ptr [esp];
FDIVP st(1), st(0);
FADDP st(1), st(0);
FSTP y;
```

Program obliczający średnią średnia z elementów 27-elementowego wektora o wartościach rzeczywistych pojedynczej precyzji.

```
MOV esi, w;
MOV ecx, 27; // rozmiar wektora
FLDZ;
petla:
FLD dword ptr [esi + 4 * ecx - 4];
FADDP st(1), st(0);
DEC ecx;
JNZ petla;
PUSH 27;
FILD dword ptr [esp];
FDIVP st(1), st(0);
ADD esp, 4;
FSTP srednia;
```

---

$Y = \text{Pi} * \log_2 10 * |(x-1)^3 + a|^{-1/2}$

```
FLDPI;
FLDL2T;
FMULP st(1), st(0);
FLD x;
FLD 1;
FSUBP st(1), st(0);
FLD st(0);
FLD st(0);
FMULP st(1), st(0);
FMULP st(1), st(0);
FLD a;
FADDP st(1), st(0);
FABS;
FSQRT;
FDIVP st(1), st(0);
FSTP y;
```

---

Oblicz przekątną prostopadłościanu o bokach rzeczywistych a, b i c:

```
FLD a;
FLD st(0);
FMULP st(1), st(0);
FLD b;
FLD st(0);
FMULP st(1), st(0);
FADDP st(1), st(0);
FLD c;
FLD st(0);
FMULP st(1), st(0);
FADDP st(1), st(0);
FSQRT;
```

Trójmian kwadratowy: To liczy gdy trójmian jest mniejszy, równy zero lub większy od zera.

```

FLD b;
FLD st(0);
FMUL;
FLD a;
FLD c;
FMUL;
FADD st, st(0);
FADD st, st(0);
FSUB;
FTST;
FSTSW ax;
SAHF;
JZ rowne;
JC mniejsze;
FSQRT;
FLD st(0);
FLD b;
FSUB;
FLD a;
FADD st, st(0);
FDIV;
FSTP Re1;
FCHS;
FLD b;
FSUB;
FLD a;
FADD st, st(0);
FDIV;
FSTP Re2;
FLDZ;
FSTP Im1;
FLDZ;
FSTP Im2;
JMP koniec;

rowne:
FLD b;
FLD a;
FADD st, st(0);
FDIV;
FCHS;
FLD st(0);
FSTP Re1;
FSTP Re2;
FLDZ;
FSTP Im1;
FLDZ;
FSTP Im2;
JMP koniec;

mniejsze:
FABS;
FSQRT;
FLD a;
FADD st, st(0);
FDIV;
FLD st(0);
FSTP Im1;
FCHS;
FSTP Im2;
FLD b;
FLD a;
FADD st, st(0);
FDIV;
FCHS;
FLD st(0);
FSTP Re1;
FSTP Re2;
JMP koniec;

koniec:

```

$y = [x_1 \dots x_n][z_1 \dots z_n]$  taka macierz

```

.
.
.
zn]

MOV ecx, n;
MOV esi, x;
MOV edi, z;
FLD [esi];
FLD [edi];
FMUL;
DEC ecx;

```

Petla1:

```

ADD esi, 8;
ADD edi, 8;
FLD [esi];
FMUL [edi];
FADD;
DEC ecx;
JNZ Petla1;

```

---


$$A = e^x = 2^{x \cdot \log_2 e}$$

```

FLD x;
FLDL2E;
FMUL;
FLD st(0);
FRNDINT;
FSUB st(1), st(0);
FXCH st(1);
F2XM1;
FLD1;
FADD;
FSCALE;
FSTP A;

```

---


$$A = x^y = 2^{y \cdot \log_2 x}$$

```

FLD y;
FLD x;
FYL2x;
F2XM1;
FLD1;
FADD;
FSTP A;

```

---