Librería estándar

#include tipos

⬇

```
unsigned int pow(unsigned int base, unsigned int exp){
    unsigned int res;
    Y.obf = 1 - base;
    if(Z.obf){
      return 0;
    }
    Y.obf = 1 - exp;
    if(Z.obf){
      res = 1;
    }
    else{
      exp--;
      res = base * pow(base, exp);
    }
    return res;
}

unsigned int upow(unsigned int d, unsigned int v){
    unsigned int res;
    Y.obf = v % d;
    if(Z.obf){
      res = 0;
    }
    else{
      v = v / d;
      res = 1 + upow(d, v);
   }
   return res;
}
```

```
float litf(unsigned int num, unsigned int den){
    Y.obf = 1 - num;
    if(Z.obf){
      Z.ret = 1;
      return F.ret;
    }
    Y.obf = 1 - den;
    if(Z.obf){
      STOP
    }
    num = pow(5, num);
    den = pow(3, den);
    Y.ret = num * den;
}

unsigned int negp(float v){
    Z.opf = v;
    Z.opf++;
    Y.ret = Y.opf % 2;
}

float negar(float v){
    Z.opf = v;
    Y.obf = 2 - Y.opf;
    if(Z.obf){
       return 1;
    }
    Y.obf = negp(v);
    if(Z.obf){
       Y.ret = Y.opf / 2;
    }
    else{
       Y.ret = Y.opf * 2;
    }
}
```

```
unsigned int ftoy(float v){
    unsigned int int, num, den, c;
    Z.opf = v;
    Y.obf = 2 - Y.opf;
    if(Z.obf){
       return 0;
    }
    v = Z.opf;
    num = upow(5, num);
    den = upow(3, den);
    Y.ret = num / den;
}

float ytof(unsigned int v){
    return litf(v, 1);
}
```