O in CODE stands for OBFUSCATION

Let's code 4 fun this time and forget about the business sides of code

DASKOM 1337

WHO AM I ?

Just another coder, having too much spare time to code

Fakhri || fai || f4r4w4y

TABLE OF CONTENTS

DASKOM 1337

“Coding is not just about creating a virtual business, it’s also be done for the sake of fun and happiness”

—no one

01

INTRODUCTION

[d( )] DASKOM 1337

So, what is code ?

Code is a language that human use to interact with a machine in order for the machine to understand what the human want the machine to do

DASKOM 1337

Obfuscation and Golfing

Obfuscation is about changing a code to make it kinda confusing for even a human to read it

```python
# Real Code (Readable)
print('Hello World')

# Obfuscated Code (Unreadable)
print((a:=''.join([str(chr(int(f'0x{p}',
0))+(' ' if o==3 else '')) for o,p in
enumerate('0x480x650x6c0x6f0x570x720x64'
.split('0x')[1:])]))+f'\r{a[:3]}l{a[3:6]
}o{a[6:7]}l{a[7:8]}')
```

Golfing in the other hand is about crafting a code as short as possible in order to achieve something

```python
# Real Code (58 Bytes)
if a:=input()=='0':
   print('yes')
else:
   print('no')

# Golfed Code (33 Bytes)
print('yneos'[(input()!='0')::2])
```

DASKOM 1337

The Function

Obfuscation is meant:

- For fun
- For slowing down reverse engineering process (by making the code harder to understand)

Golfing is meant:

- For fun
- For minifying the code to save some space for memory

02

The Techniques

(Mainly, using C & python Language)

DASKOM 1337

Type Casting

Instead of typing the character directly,
we can convert it to other type first

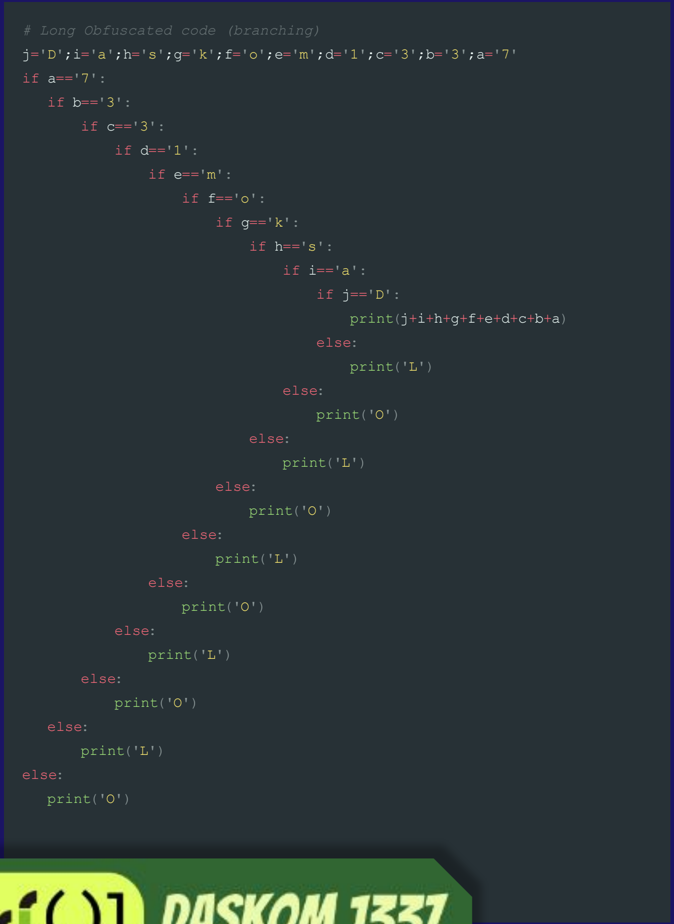
```python
import binascii

# Real Code
print('Daskom1337')

# Obfuscation process (hex mode)
print(binascii.hexlify(b'Daskom1337')) # -> 4461736b6f6d31333337
# Obfuscated code (hex mode)
print(bytearray.fromhex('4461736b6f6d31333337'))

# Obfuscation process (string -> ascii -> hex)
print(''.join([hex(ord(x)) for x in 'Daskom1337'])) # -> 0x440x610x730x6b0x6f0x6d0x310x330x330x37
# Obfuscated code (string -> ascii -> hex)
print(''.join([chr(int(f'0x{x}', 0)) for x in '0x440x610x730x6b0x6f0x6d0x310x330x330x37'.split('0x')[1:]]))
```

Unnecessary Branching

Add branching to make the code look more complex,
thus harder to read

```
# Real Code
print('Daskom1337')

# Obfuscated code (branching)
print((j:='D' if (i:='a' if (h:='s' if (g:='k' if (f:='o' if (e:='m' if (d:='1' if (c:='3' if (b:='3' if
(a:='7')=='7' else 'L')=='3' else 'O')=='3' else 'L')=='1' else 'O')=='m' else 'L')=='o' else 'O')=='k' else
'L')=='s' else 'O')=='a' else 'L')+i+h+g+f+e+d+c+b+a)


# Longer one -> next slide (cause its really long XD)
```

DASKOM 1337



```
# Long Obfuscated code (branching)
j='D';i='a';h='s';g='k';f='o';e='m';d='1';c='3';b='3';a='7'
if a=='7':
   if b=='3':
       if c=='3':
           if d=='1':
               if e=='m':
                   if f=='o':
                       if g=='k':
                           if h=='s':
                               if i=='a':
                                   if j=='D':
                                       print(j+i+h+g+f+e+d+c+b+a)
                                   else:
                                       print('L')
                               else:
                                   print('O')
                           else:
                               print('L')
                       else:
                           print('O')
                   else:
                       print('L')
               else:
                   print('O')
           else:
               print('L')
       else:
           print('O')
   else:
       print('L')
else:
   print('O')
```

See,
it's pretty looooooooooooooooooooooooong right ?

This is really unnecessary just
like this line over there :v

Unnecessary Looping

Just like branching, but indeed it's a loop

```
# Real Code
print('Daskom1337')

# Obfuscated code (looping)
print(''.join([chr(i).upper() for i in range(ord('a'),ord('z'))][3:4]+[chr(i) for i in
range(ord('a'),ord('z'))][0:1]+[chr(i) for i in range(ord('a'),ord('z'))][18:19]+[chr(i) for i in
range(ord('a'),ord('z'))][10:11]+[chr(i) for i in range(ord('a'),ord('z'))][14:15]+[chr(i) for i in
range(ord('a'),ord('z'))][12:13]+[str(i) for i in range(10)][1:2]+[str(i) for i in range(10)][3:4]+[str(i) for i
in range(10)][3:4]+[str(i) for i in range(10)][7:8]))


# Longer one (more readable version) -> next slide
```

DASKOM 1337

```python
# Long Obfuscated code (looping)
print(''.join([chr(i).upper() for i in range(ord('a'),ord('z'))][3:4]
           + [chr(i) for i in range(ord('a'),ord('z'))][0:1]
           + [chr(i) for i in range(ord('a'),ord('z'))][18:19]
           + [chr(i) for i in range(ord('a'),ord('z'))][10:11]
           + [chr(i) for i in range(ord('a'),ord('z'))][14:15]
           + [chr(i) for i in range(ord('a'),ord('z'))][12:13]
           + [str(i) for i in range(10)][1:2]
           + [str(i) for i in range(10)][3:4]
           + [str(i) for i in range(10)][3:4]
           + [str(i) for i in range(10)][7:8]
))
```

DASKOM 1337

Golfing

Either obfuscation or golfing shares the same technique which is odd right ?, it surely is…

But the main difference in golfing is that

Our Target is to make the code **as short as possible**, which turns out to be pretty challenging, and this will burn our brain so hard after doing it quite a while 🤯

Macros and Ternary operator

We can use C Macros to golf a code,
Just like this :

```
#include<stdio.h>
#define a printf
#define b "Daskom1337"
main() {a(b);}
```

Also there is this thing called "Ternary operator" which is actually a one line if function using "?" and ":"

```
main(){
// Real code (70 bytes)
char c = getchar();
if(c=='Y') {
   a("Yes");
} else {
   a("No");
}

// Golfed code (29 bytes)
a(getchar()=='Y'?"Yes":"No");
}
```

[d( )] DASKOM 1337

References

Python

- Code Golfing in Python
- Tips for golfing in Python

C

- Tips for golfing in C
- https://www.codingame.com/forum/t/tips-and-tricks-for-code-golfing-in-c/694/7
- How to C-Golf

CODE GOLFING AND OBFUSCATION COMPETITION

DASKOM 1337

COMPETITION

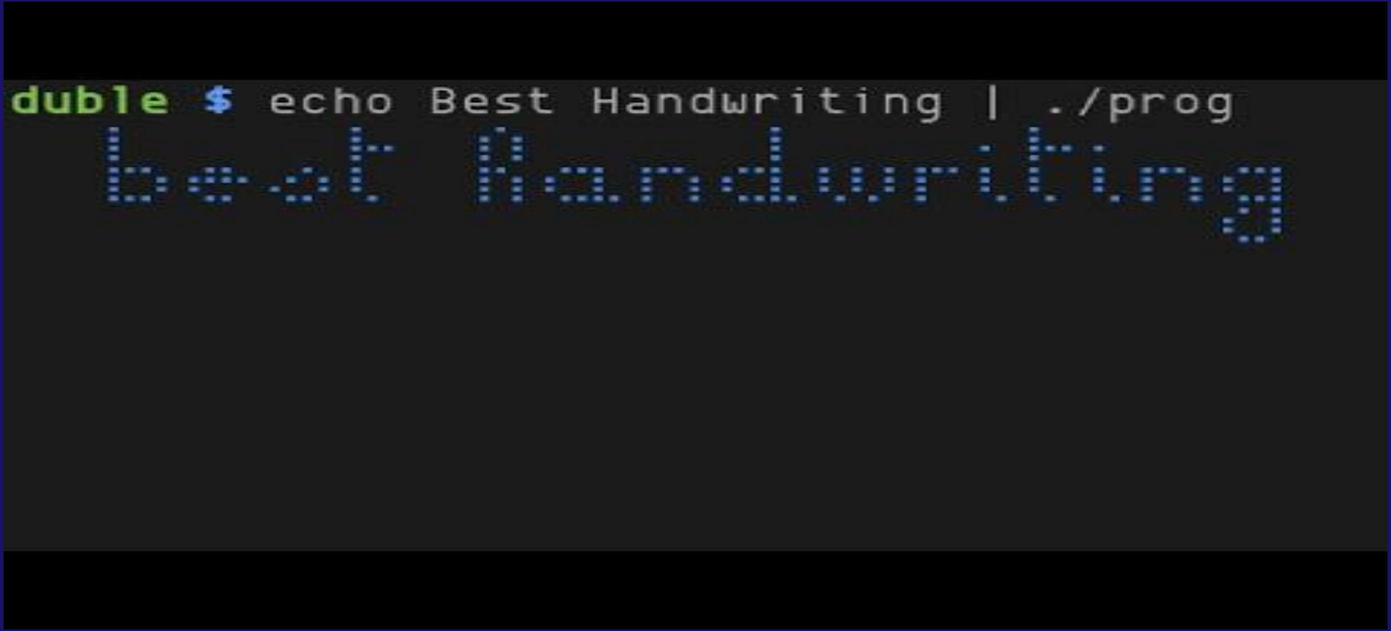
- The International Obfuscated C Code Contest
- Code Golf
- JS1k - The JavaScript code golfing competition

03

EXAMPLE(S)

DASKOM 1337

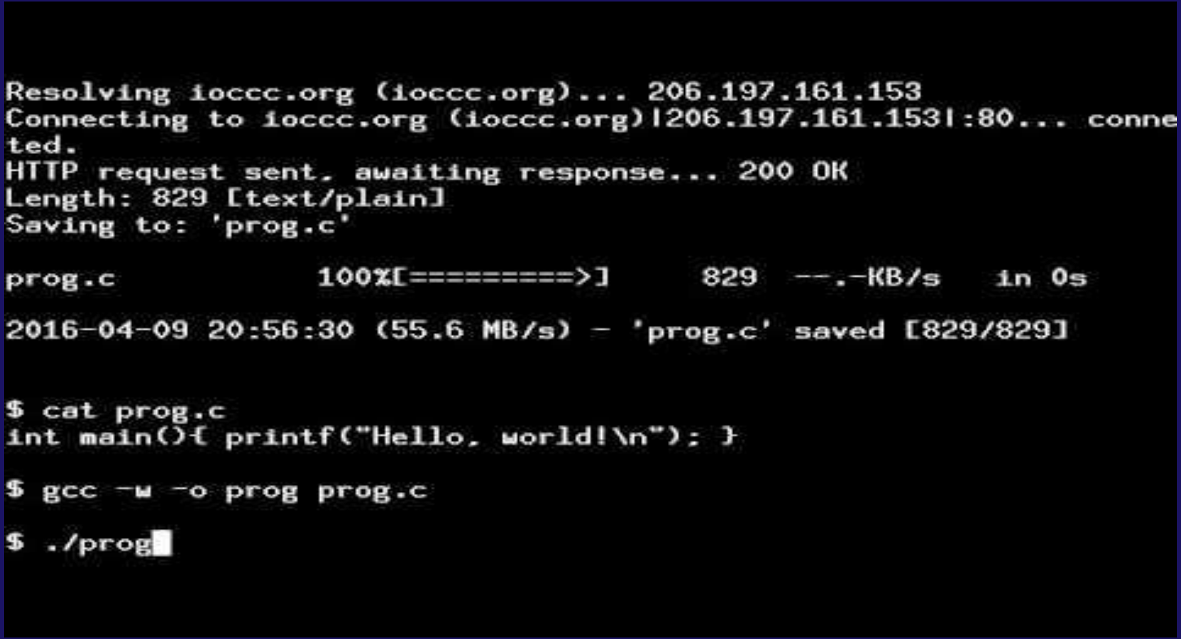
Best Handwriting, using Braille, IOCCC 2015 winner



```
duble $ echo Best Handwriting | ./prog
```

DASKOM 1337

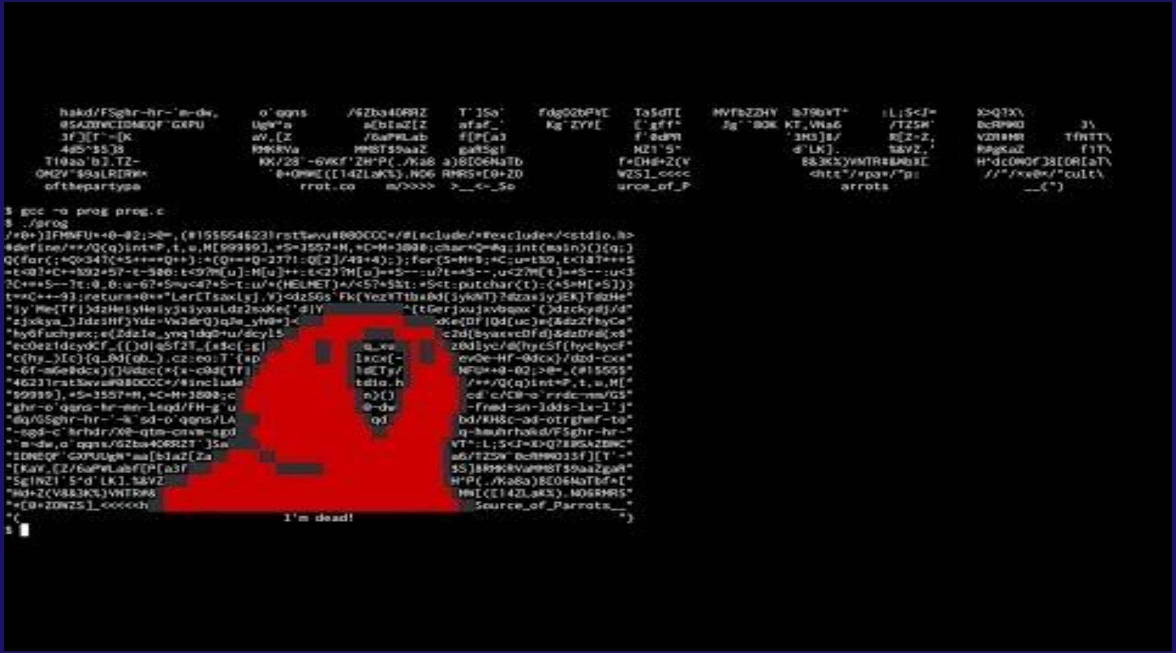
Most Overlooked Obfuscation – IOCCC 2015 endoh2.c



```
Resolving ioccc.org (ioccc.org)... 206.197.161.153
Connecting to ioccc.org (ioccc.org)|206.197.161.153|:80... conne
ted.
HTTP request sent, awaiting response... 200 OK
Length: 829 [text/plain]
Saving to: 'prog.c'

prog.c              100%[=========>]     829  --.-KB/s   in 0s

2016-04-09 20:56:30 (55.6 MB/s) - 'prog.c' saved [829/829]


$ cat prog.c
int main(){ printf("Hello, world!\n"); }

$ gcc -w -o prog prog.c

$ ./prog
```

DASKOM 1337

Best use of python - IOCCC 2018 endoh2.c



```
    hakd/FSghr-hr-`m-dw,      o`qqns     /6Zba4ORRZ    T`]Sa`    fdgO2bPYE    TaSdT[     MVfbZZHY  bJ9bYT*      :L;S<J=       X>Q7X\
    @SAZBNCIDNEQF`GXPU       UgM"a          a[bIaZ[Z    afaf_`     Kg`ZYY[     [`gff"      Jg``BOK KT,VNa6       /TZSM`        BcRMMO          J\
    3f][T`-[K                aV,[Z          /6aPWLab    f[P[a]                  f`0dPR                `3M3]B/       R[Z-Z,        VIR#MR      TfNTT\
    4d5^$5]8                 RMKRVa        MMBT$9aaZ    gaRSg1                  MZ1`5"                d`LK].        %&VZ,'        RAgKaZ         f1T\
   T10aa`b].TZ-               KK/28`-6VKf`ZH"P(./Ka8 a)8[O6NaTb             f+[Hd+Z(V              B&3K%)VNTR#&Mb#[      H*dcOMOf]8[OR[aT\
  OM2V"$9aLR[RW*               `0+OMM[([14ZLaK%).NO6 RMRS+[0+ZD             WZS1_<<<<                <htt"/*pa*/"p:        //"/*e@*/"cult\
   ofthepartypa                   rrot.co    m/>>>>  >__<-_So               urce_of_P                    arrots                 __(")
$ gcc -o prog prog.c
$ ./prog
/*0+)]FMNFU*+0-02;>@=,(#155554623!rst%evu#000CCC*/#include/*#exclude*/<stdio.h>
#define/**/Q(q)int*P,t,u,M[99999],*S=3557+M,*C=M+3000;char*Q=#q;int(main)(){q;}
Q(for(;*Q>34?(*S++=*Q++):*(Q+=*Q-27?1:Q[2]/49+4););for(S=M+9;*C;u=t%9,t<18?***S
=t<0?*C++%92*57-t-500:t<9?M[u]:M[u]++:t<27?M[u]=*S--:u?t=*S--,u<2?M[t]=*S--:u<3
?C+=*S--?t:0,0:u-6?*S=u<4?*S-t:u/*(HELMET)*/<5?*S%t:*S<t:putchar(t):(*S=M[*S]))
t=*C++-9];return+0**"Ler[Tsaxlsj.Y}<dzSGs`Fk(Yez#Ttb+0d(iyKNT)?dzaxiyjEK}TdzHe"
"iy`Me{Tf|)dzHeiyHeiyjxiyaxLdz2mxKe(`d|Y            ^{tGerjxujxvbqqx`()dzckydj/d"
"zjxkya_)JdziHf}Ydz-Vw2drQ)qJe_yh0*]<             xKe{Df|Qd[uc)e{&dzZfhyCe"
"hy6fuchyex;e(Zdz1e_ymq1dqD+u/dcyl5             c2d[byaxecDfd]&dzDYd(x6"
"ecOez1dcydCf_{[()d|qSf2T_{x$c[;g|     u_xe       z0dlyc/d(hycSf(hychycF"
"c(hy_)Ic}{q_0d(qb_).cz:eo:T`{*p      1xcx[-      evOe-Hf-0dcx}/dzd-cxe"
"-6f-m6e0dcx){]Udzc(*{x-c0d(Tf|       hd[Ty/      NFU*+0-02;>@=,(#15555"
"4623]rst%evu#000CCC*/#include        tdio.h      /**/Q(q)int*P,t,u,M["
"99999],*S=3557+M,*C=M+3000;c          n){}       cd`c/C0-o`rrdc-mm/GS"
"ghr-o`qqns-hr-mr-1eqd/FH-g`u          0-dw       -fnmd-sn-1dds-1x-1`j"
"dq/GSghr-hr-`-k`sd-o`qqns/LA          qd         bd/KH&c-ad-otrghmf-to"
"-sgd-c`hrhdr/X0-qtm-cnvm-sgd                      q-hmuhrhakd/FSghr-hr-"
"`m-dw,o`qqns/6Zba4ORRZT`]Sa                      VT*:L;S<J=X>Q7X0SAZBNC"
"IDNEQF`GXPUUgM"aa[bIaZ[Za                        a6/TZSW`BcRMMO13f][T`-"
"[KaV,[Z/6aPWLabf[P[a3f                           $S]8RMKRVaMMBT$9aaZgaR"
"SgIMZ1`5"d`LK].%&VZ                              H"P(./Ka8a)8[O6NaTbf+["
"Hd+Z(V8&3K%)VNTR#&                               MN[([14ZLaK%).NO6RMRS"
"+[0+ZDNZS]_<<<<h                                 Source_of_Parrots__"
"(                       I'm dead!                                    ")
$ █
```

DASKOM 1337

04

Winner Showcase

🏆🏆🏆

[d( )] DASKOM 1337

DRUMROLL...

DASKOM 1337

ADRAMA (75 bytes)

```
a=list(iter(input,''));print((len(a)-2!=len(a[0])//2)*"Non "+"Symmetrical")
```

~~🏆~~

Week 12 Champion

[c( )] DASKOM 1337

THANKS !

Ask me anything !
I'll answer as best as i can