

How are you feeling
about the exam?

CS 340

Exam 1 Review

clicker.cs.illinois.edu

Q1

~Code~
340



Updates

1. Exam 1 - Next Tuesday, September 23rd
 - a. **Sign up now!** 
 - b. **Study guide & Practice Exam out** 
 - c. **No class next Tuesday (23rd)** 
 - d. **Submit DRES to CBTF Directly** 
2. MP3 - Due Tuesday September 30th 

Agenda

1. Review

- a. Bits, bytes and data
- b. Coding in C ←
- c. Hardware ←

Big Numbers Correction

Value	base-10	Prefix	Pronounced
2^{10}	1024	Ki	<u>Kilo</u>
2^{20}	1,048,576	Mi	<u>Mega</u>
2^{30}	1,073,741,824	Gi	Giga
2^{40}	1,099,511,627,776	Ti	Tera
2^{50}	1,125,899,906,842,624	Pi	Peta
2^{60}	1,152,921,504,606,846,976	Ei	Exa

10^3
 10^6

Bit: 1 or 0

Byte: 8 bits, 0-255
↓ decimal

Decimal: base-10 0-9

Binary: base-2 0, 1

Hexadecimal: base-16 0-9, A-F
↑ 2 digits
0-15 ↓ dec

How many bits do I need to represent the decimal number 513?

1	0	0	0	0	0	0	0	0	1	
10	9	8	7	6	5	4	3	2	1	
1024	512	256	128	64	32	16	8	4	2	1

10

clicker.cs.illinois.edu

Q2

~Code~
340



What is 0x134 in binary?

Handwritten diagram showing the conversion of 0x134 to binary:

0x134 is split into three 4-bit segments: 0001, 0011, and 0100. The label "4 bits" is written above the third segment.

0001_0011_0100

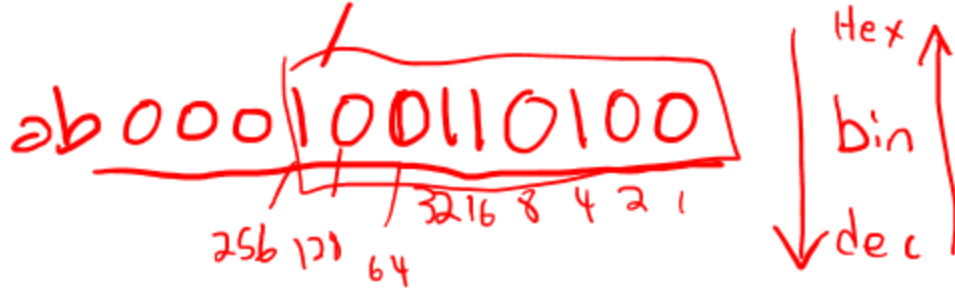
clicker.cs.illinois.edu

Q3

~Code~
340



What is 0x134 in decimal?



clicker.cs.illinois.edu

Q4

~Code~
340



$$4 + 16 + 32 + 256 = 308$$

Data Types Example 1

char x = 'k'; $\rightarrow 107$
 $0x6B$

X
 $0x6B$

$0x1001$

little endian

y
34 01 00 00 $\rightarrow ?$
03 04 05 06

$0x1002$

int *p = &y;
8 bytes

P

02 10 00 00 00 00 00 00
01 A B C D E F 10

$0x1008$

$0x0000000000000002$

How is the char 'b' stored in big endian?

0x → Hex

0b → binary

0x62 → 0b01100010

How big is a char? → 1 byte
2 Hex digits

0x62

98 ← dec

clicker.cs.illinois.edu

Q5

~Code~
340



**How is the char 'b' stored
in little endian?**

↪ 0x62
98 dec

0x62

clicker.cs.illinois.edu

Q6

~Code~
340

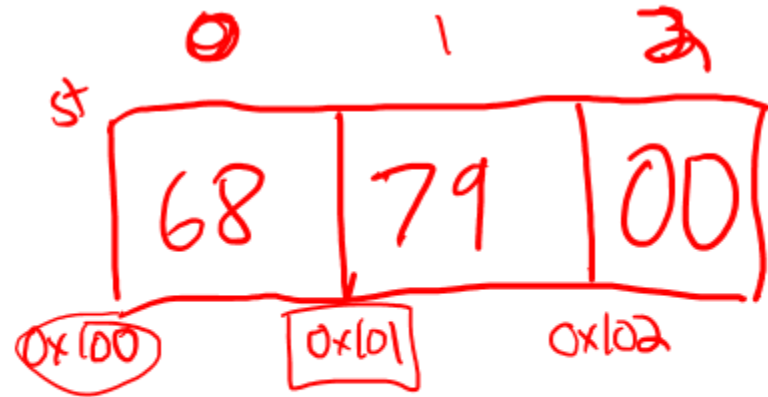


Data Types Example 2

`char st[3] = "hy"`



A diagram showing a 2D array representation of the string "hy". It consists of a 2x3 grid of boxes. The first row contains 'h', 'y', and a null terminator '\0'. The second row contains '\0', '\0', and '\0'. Below the first column, the memory address 0x68 is written. Below the second column, the memory address 0x79 is written. An arrow points from the index 0 in the code to the first column of the grid.



```
printf("%d#x", st[0]);  
printf("%c", st[0]); → 0x68
```

What prints in hexadecimal?

clicker.cs.illinois.edu

Q7

~Code~
340



12
13
14

```
char str[5] = "aaaa"  
int *ptr = (int*)str  
printf("%#x", *ptr);
```

Handwritten notes:
- Red arrows point to `str[5]` in the first line and `ptr` in the third line.
- A red box highlights `"aaaa"` in the first line.
- A red box highlights `(int*)str` in the second line.
- A red box highlights `*ptr` in the third line.
- A red arrow points from `ptr` to `*ptr`.
- A red arrow points from `ptr` to the memory diagram below.
- A red arrow points from `str[5]` to the memory diagram below.
- A red arrow points from `ptr` to the memory diagram below.

'a' = 97

str[5] = 'a';

1. what is 'a' in binary? 0110 0001

2. what is 'a' in hex? 0x61

3. How is this data stored?

4. line 13?

5. line 14?

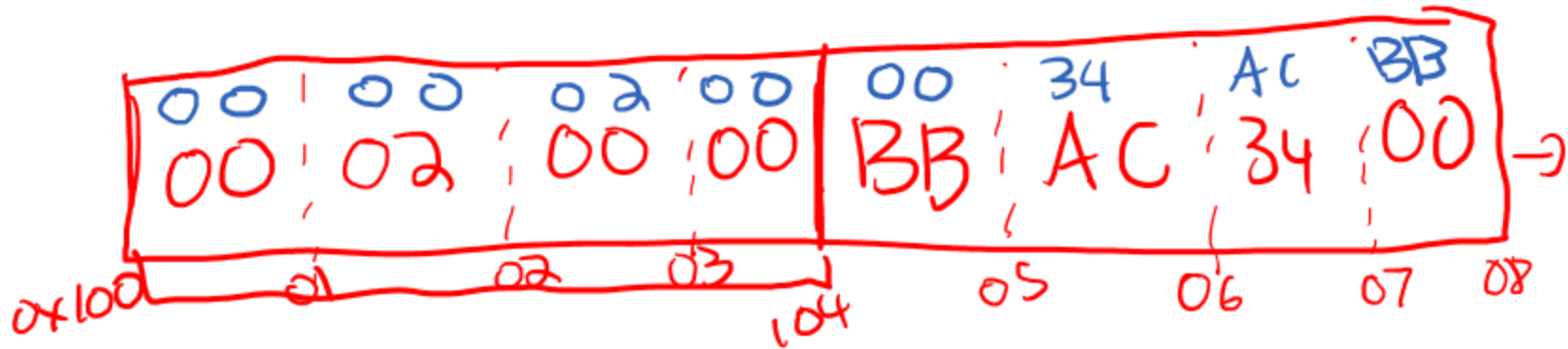


Handwritten notes:
- `61 61 61 61`
- `61 61 61 61`

Handwritten note:
`char str[3] = "aaaa";` `1 a a a`

Data Types Example 3

int arr[2] = { 512, 3452091 };
1
3
0x00 34 AC BB
✓



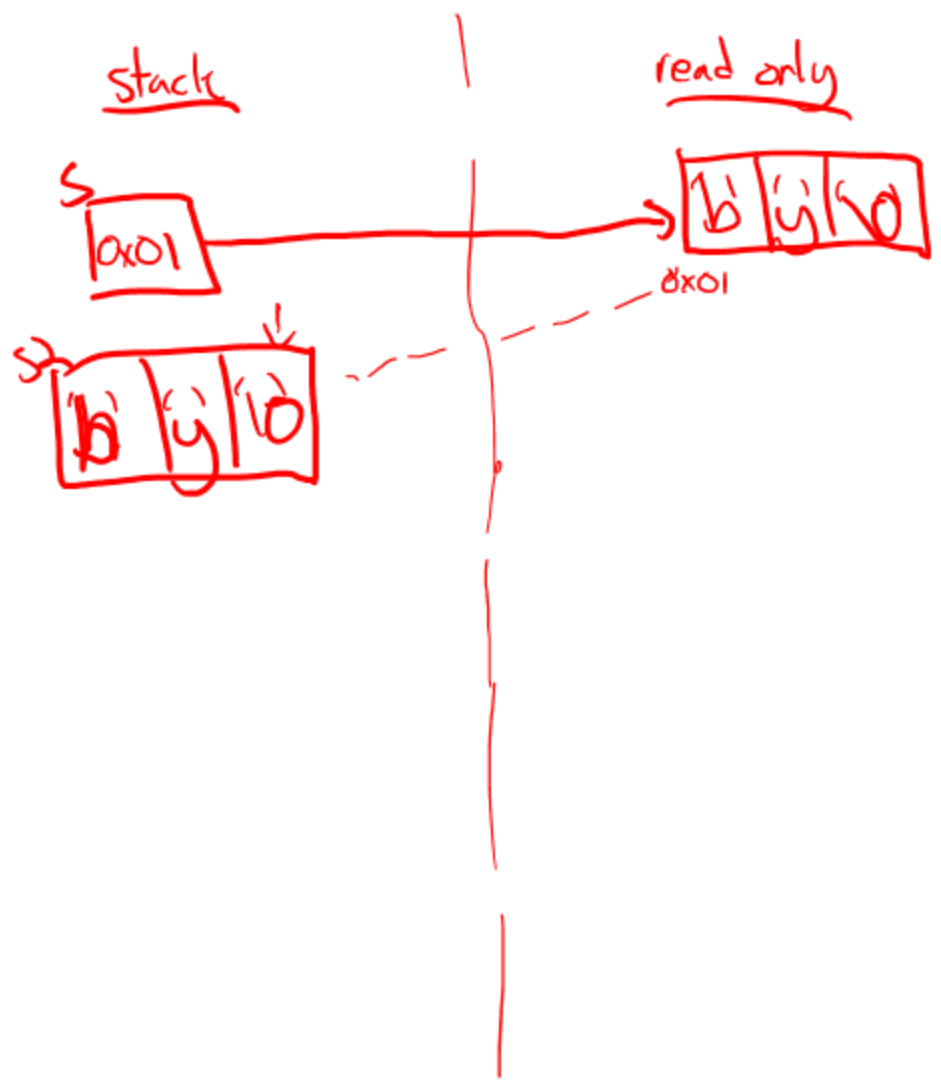
512 = 0x0200
?

Big endian

Data Types Example 4

C-Strings Example 1

```
char *s = "by";  
char s2[3] = "hy";  
strcpy(s2, s);
```



C-Strings Example 2

What goes in the blank

```
void initCombo(, char *s, char *s2);
```

clicker.cs.illinois.edu

Q8

~Code~
340



Do I need to use strcpy?

```
void initCombo(combo *self, char *s1, char *s2){  
    self->word1 = s1;  
    self->word2 = s2;  
}
```

clicker.cs.illinois.edu

Q9

~Code~
340



What does this code do?

```
int main() {  
    FILE *fl = fopen("file.dat", 'rb');  
    int bufi;  
    char bufc[4];  
    → fread(&bufi, 4, 1, fl);  
    → fseek(fl, 0, SEEK_SET);  
    → fread(&bufc, 4, 1, fl);  
}
```

bufi
→

GB	54	12	6B
----	----	----	----

bufc

GB	54	12	6B
----	----	----	----

GB	54	12	6B
----	----	----	----

 17 00 89

little
0xGB12546B

Big
0xGB54126B

What happens?

```
14 char *createWord(char a, char b){  
15     char wrd[3] = {a, b, '\0'};  
16     return wrd;  
17 }  
18  
19 int main() {  
20     char *wrd = createWord('a', 'b');  
21 }
```

clicker.cs.illinois.edu

Q10

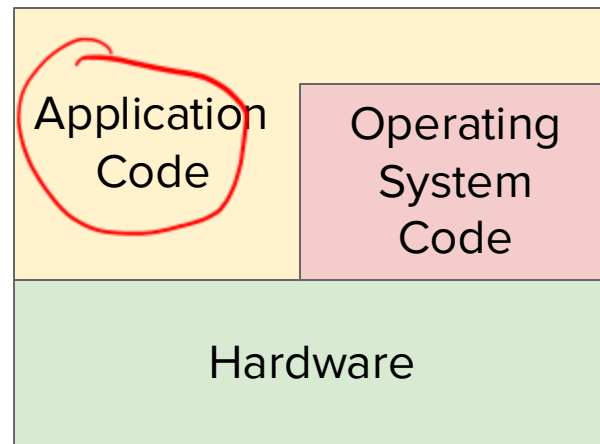
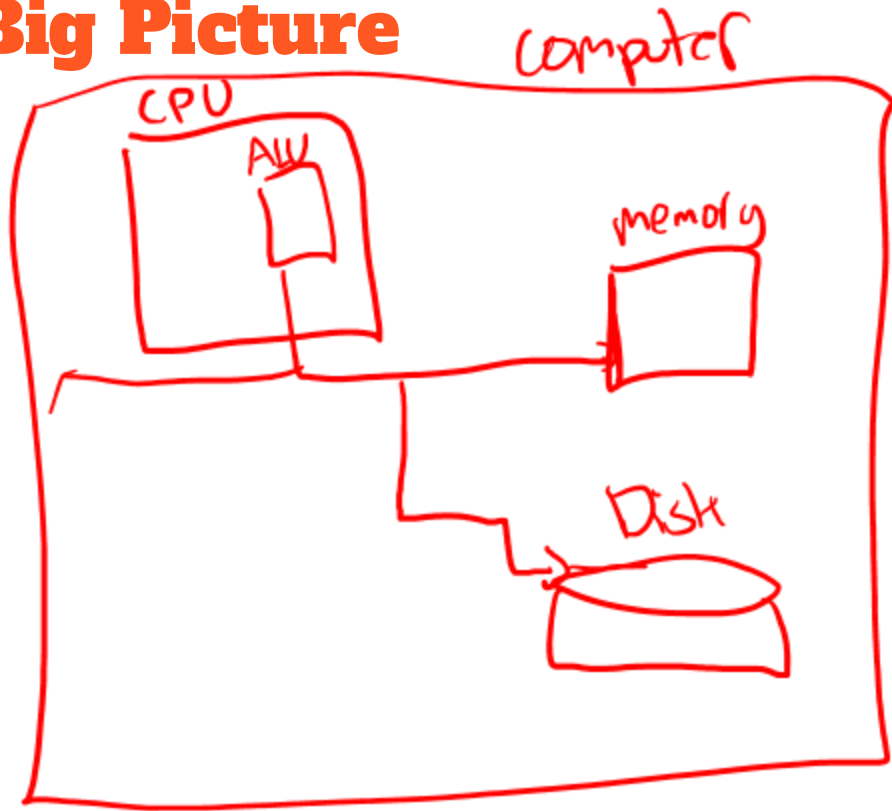
~Code~
340



```
14 ▾ char *createWord(char a, char b){  
15     char *wrd = malloc(3);  
16     wrd[0] = a;  
17     wrd[1] = b;  
18     wrd[2] = '\\0';  
19     return wrd;  
20 }  
21  
22 ▾ int main() {  
23     char *wrd = createWord('a', 'b');  
24     free(wrd);  
25 }
```

Hardware

Big Picture



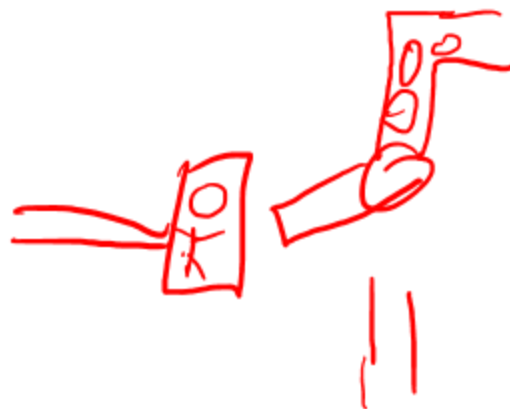
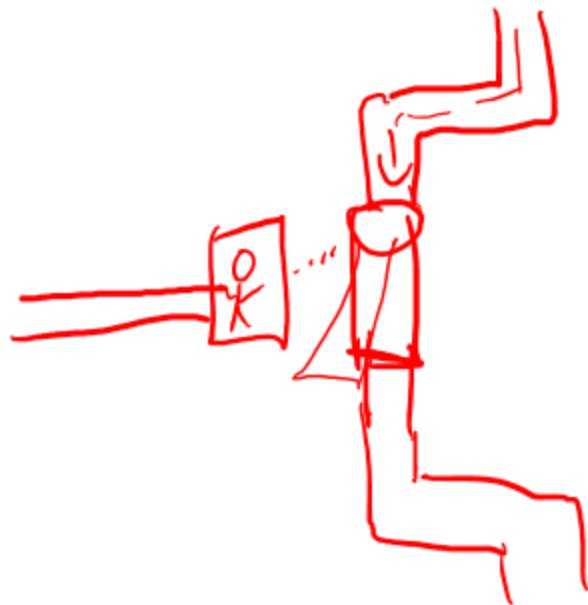
Transistors



PMOS
default ~~closed~~
connected



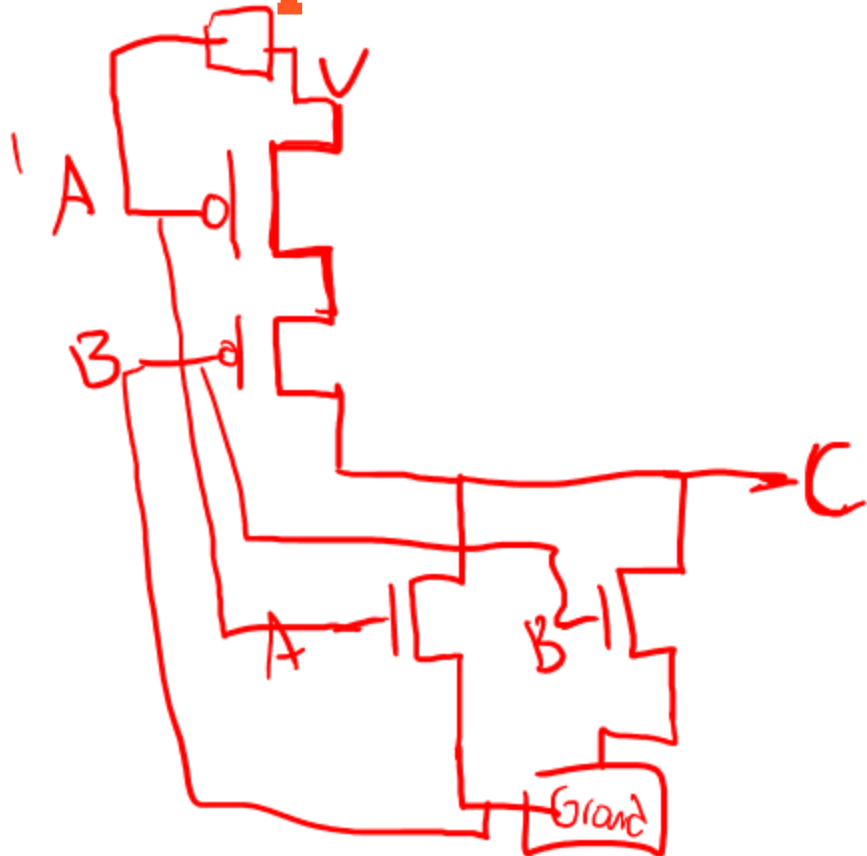
NMOS
default open



Gate Example

$$A \text{ NOR } B = C$$

A	B	C
1	0	0
0	1	0
0	0	1
1	1	0



1-bit Adder

$$x_0 + y_0 = z_1 z_0$$

$$1 + 1 = 10$$

$$0 + 1 = 01$$

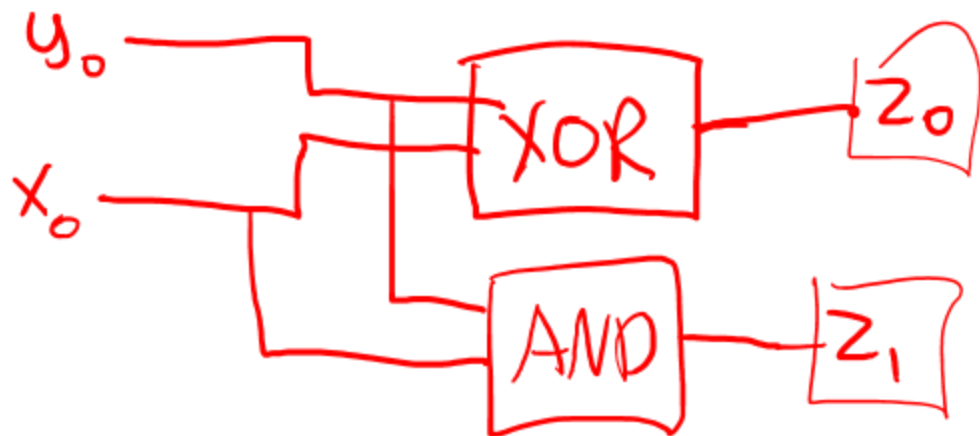
$$0 + 0 = 00$$

$$1 + 0 = 01$$

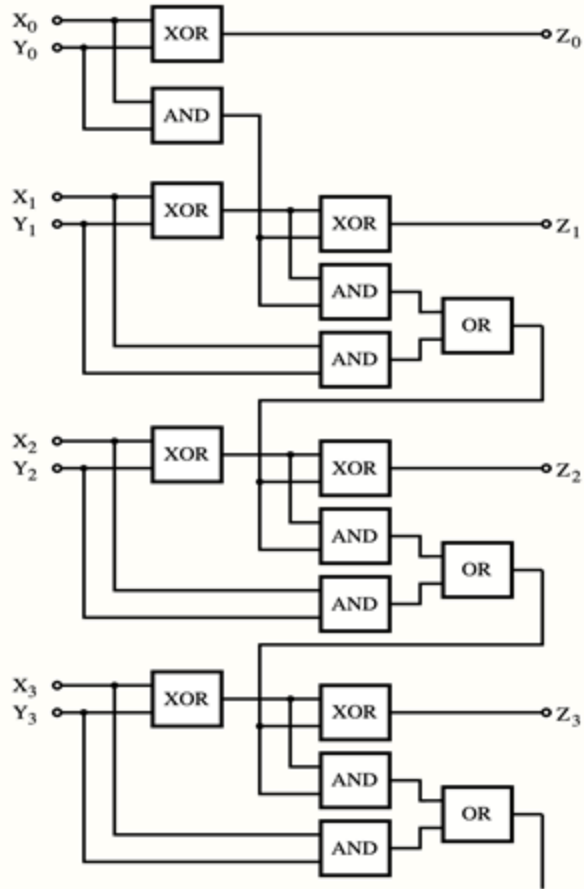
$$Z_0 = x_0 \overset{\text{XOR}}{\wedge} y_0$$

$$Z_1 = C_1$$

$$C_1 = y_0 \& x_0$$



Multi-bit Adder



What depth of 2-MUX's for a 5 input selection?

clicker.cs.illinois.edu

Q11

~Code~
340



Caching

Example of Spatial or Temporal?

```
get_key(gd_GIF *gif, int key_size, uint8_t *sub_len, uint8_t *shift, uint8_t *byte)
{
    int bits_read;
    int rpad;
    int frag_size;
    uint16_t key;

    key = 0;
    for (bits_read = 0; bits_read < key_size; bits_read += frag_size) {
        rpad = (*shift + bits_read) % 8;
        if (rpad == 0) {
            /* Update byte. */
            if (*sub_len == 0) {
                read(gif->fd, sub_len, 1); /* Must be nonzero! */
                if (*sub_len == 0)
                    return 0x1000;
            }
            read(gif->fd, byte, 1);
            (*sub_len)--;
        }
        frag_size = MIN(key_size - bits_read, 8 - rpad);
        key |= ((uint16_t) ((*byte) >> rpad)) << bits_read;
    }
}
```

clicker.cs.illinois.edu

Q12

~Code~
340



Any feedback for us?



<https://forms.gle/hk8kKjPRfLrowpwX6>