

EE 200 Lecture 2: Reading code + types

Steven Bell
8 September 2022



Office hours

- Mondays 4:30-6:30pm
- Wednesdays 10am-12pm
- Other times by appointment

Campuswire

campuswire.com/c

```
int isPositive(int x)
{
    return(x >= 0);
}
int makePositive(int x)
{
    int val = x;
    if(!isPositive(val))
        val = -val;
}
return(val);
}
int main()
{
    int x = -5;
    printf("The value is: %d\n", makePositive(x));
}
```

main

x

makePositive

x

val

isPositive

x

i++ VS ++i

```
int main(int argc, char* argv[])
{
    int i = 0;
    while(i < 5){
        printf("%d\n", i++);
    }
    printf("\n\n");
    i = 0;
    while(i < 5){
        printf("%d\n", ++i);
    }
}
```

What gets printed?

post_increment.c

```
#define PLAY 0
#define REWIND 1
#define FAST_FORWARD 2
#define STOP 3

int main(void)
{
    int mode = PLAY;
    int playing;

    switch(mode){
        case PLAY:
            playing = 1;
        case REWIND:
        case FAST_FORWARD:
        case STOP:
        default:
            playing = 0;
    }
}
```

music.c

```
if(playing){
    printf("start the music!\n");
}
else{
    printf("stop the music!\n");
}

return(0);
}
```

What gets printed?

enum

Creates a new type with a restricted set of named values

```
#define PLAY 0
#define REWIND 1
#define FAST_FORWARD 2
#define STOP 3

int main(void)
{
    int mode = PLAY;
    int playing;

    switch(mode){
        case PLAY:
            playing = 1;
        case REWIND:
        case FAST_FORWARD:
        case STOP:
        default:
            playing = 0;
    }
}
```

```
enum mode_t {
    PLAY,
    REWIND,
    FAST_FORWARD,
    STOP
};

void main(void)
{
    mode_t mode = REWIND;
    int playing;

    switch(mode){
        case PLAY:
            playing = 1;
        case REWIND:
        case FAST_FORWARD:
        case STOP:
        default:
            playing = 0;
    }
}
```

continue

```
const int THRESHOLD = 1000;
int doStuffToBigOnes(int values[], int len)
{
    for(int i = 0; i < len; i++){
        if(values[i] < THRESHOLD){
            continue;
        }
        // Do lots of stuff with the value, now that we know it's big
    }
}
```



```
bool checkIt(int thingToCheck)
{
    int test1 = 0;
    int test2 = 0;
    if(thingToCheck == 5)
        test1 = 0;
        test2 = 1;
    if(thingToCheck == 7)
        test1 = 1;
    return(test1 || test2);
}
```

What does this return for various values of `thingToCheck()` ?

Bad code in the wild

```
static OSStatus
SSLVerifySignedServerKeyExchange(SSLContext *ctx, bool isRsa, SSLBuffer signedParams,
                                uint8_t *signature, UInt16 signatureLen)
{
    OSStatus      err;
    ...

    if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
        goto fail;
    ...

fail:
    SSLFreeBuffer(&signedHashes);
    SSLFreeBuffer(&hashCtx);
    return err;
}
```

From Safari SSL key verification

typedef

Creates an alternative name for a type

```
typedef int Time;  
Time now = getTime();  
Time future = now + 100;
```

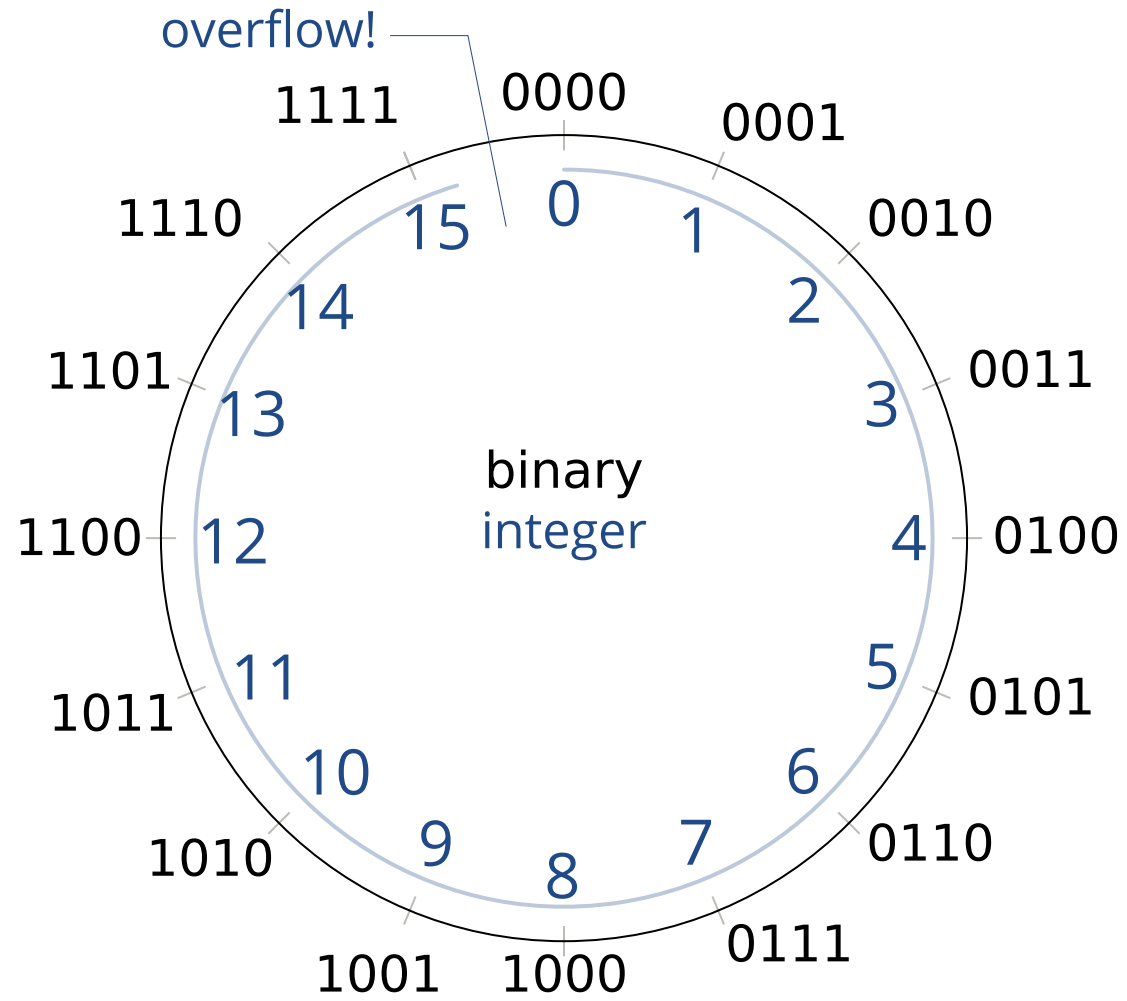
```
typedef unsigned char u8;  
typedef char s8;  
typedef short unsigned int u16;  
typedef short int s16;  
u16 myVariable = 1000;
```

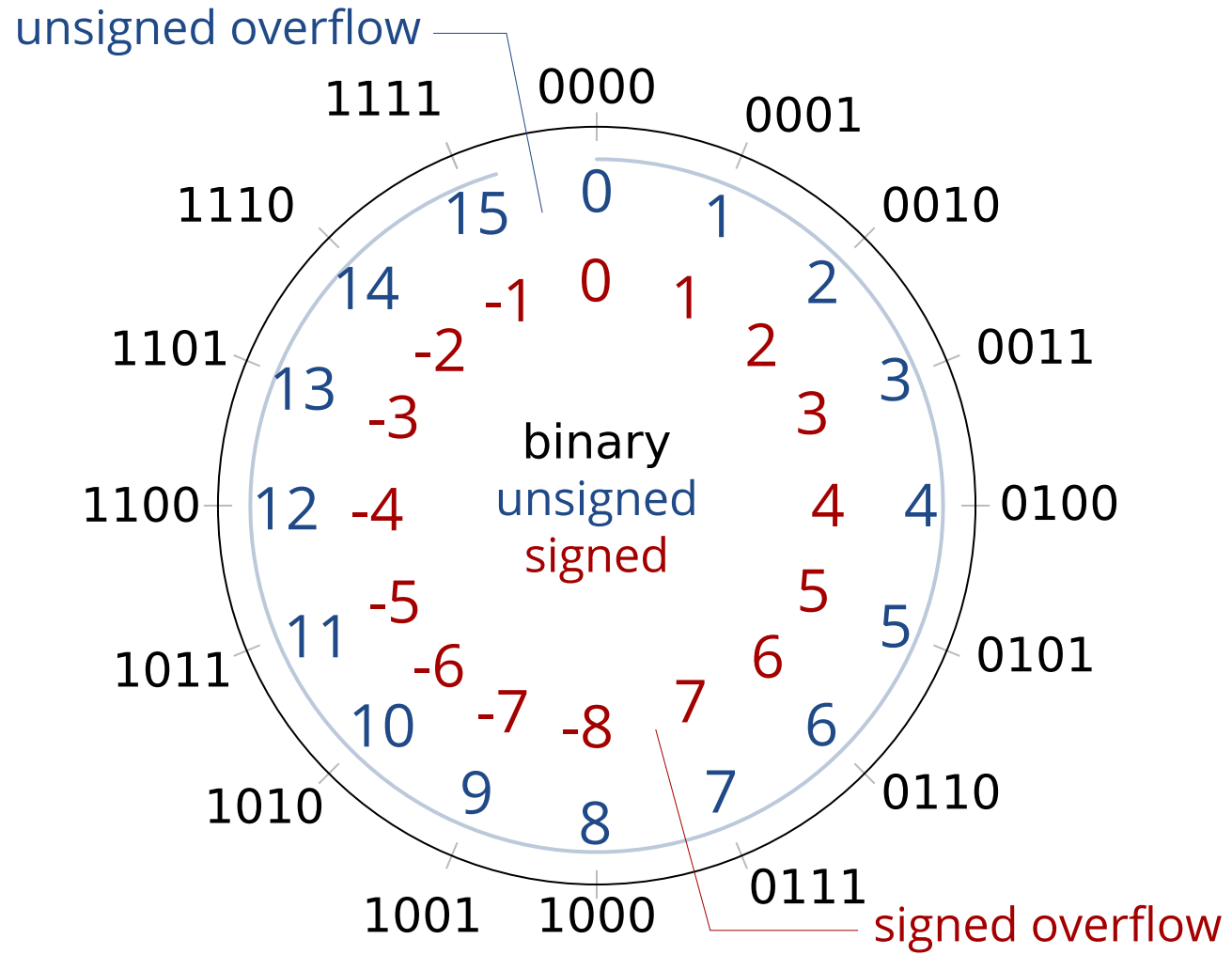
struct

Creates a type which is a bundle of related variables

```
typedef struct {  
    float x;  
    float y;  
    float z;  
} Vec3d;
```

```
Vec3d addVectors(Vec3d a, Vec3d b)  
{  
    Vec3d sum;  
    sum.x = a.x + b.x;  
    sum.y = a.y + b.y;  
    sum.z = a.z + b.z;  
    return(sum);  
}
```





To write a negative number in 2's complement:

Write the positive number in binary

Flip all the bits ($1 \rightarrow 0$, $0 \rightarrow 1$)

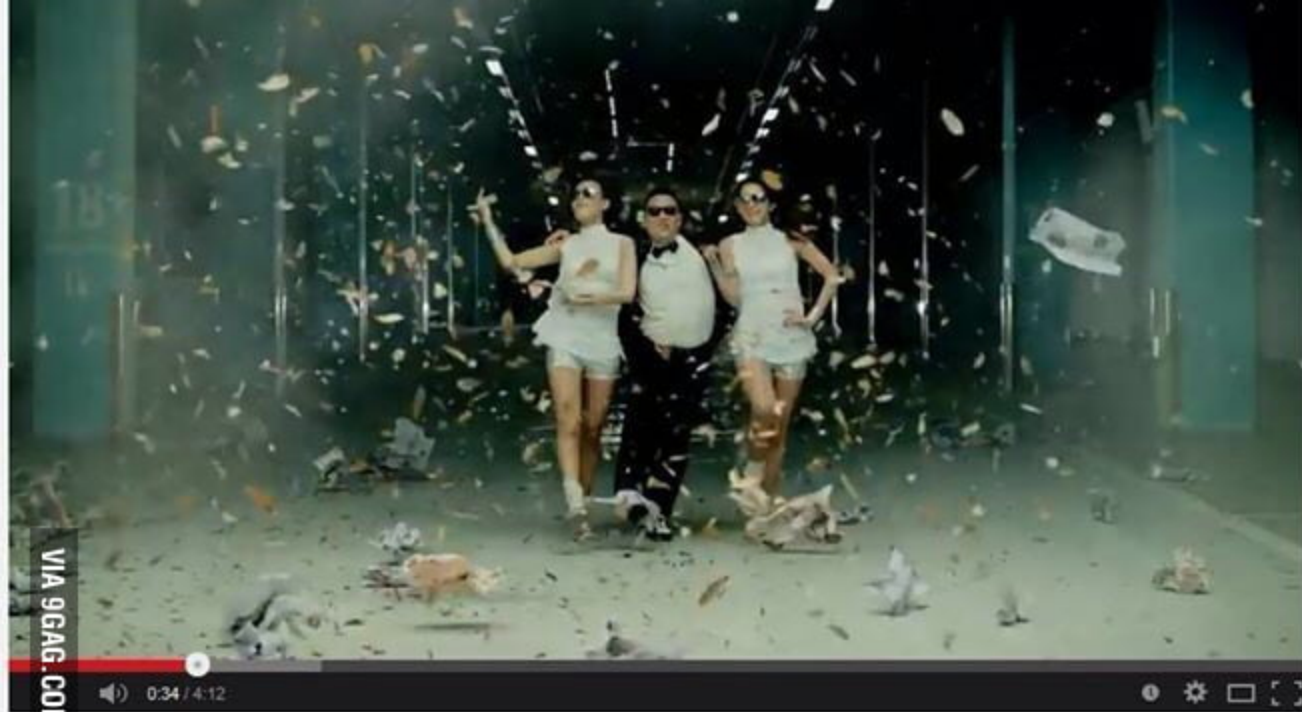
Add 1 (with all the appropriate carries)

To convert negative 2's complement to decimal,

Flip all the bits ($1 \rightarrow 0$, $0 \rightarrow 1$)

Add 1 (with all the appropriate carries)

Write the number in decimal



PSY - GANGNAM STYLE (강남스타일) M/V



officialpsy

Subscribe 7,603,314

-2142153076

+ Add to Share More

8,773,116 1,141,536



Mike Brzozowski via Google+ 3 hours ago

Gangnam Style has been watched more than MAX_INT32 times (that's well over 2 billion views). Google nerds have been eagerly anticipating this day...



YouTube originally shared this

We never thought a video would be watched in numbers greater than a 32-bit integer (=2,147,483,647 views), but that was before we met PSY. "Gangnam Style" has been viewed so many times we had to upgrade to a 64-bit integer (9,223,372,036,854,775,808)!

Hover over the counter in PSY's video to see a little math magic and stay tuned for bigger and bigger numbers on YouTube.

Bad code makes things blow up



(wikipedia)

Reused but untested

~~Bad~~ code makes things blow up
very expensive



(wikipedia)

Comparing floating-point numbers

```
float x = 9;
float y = (x / 3.0) + 2.5;
float z = (3.0 * y) - 7.5;
if(z == x) {
    printf("it's equal!\n");
}
else {
    printf("surprise! The value is %.20f, but the result was %.20f (%f)\n",
        x, z, (x-z));
}
```

Doing math with integers

What is printed as a result of running the code below?

```
int total = 350;
int dailyAveragePercent = total / 365 * 100;
printf("average (percentage): %d\n", dailyAveragePercent);
```

ProTip

- Use up arrow to repeat a command in the shell
- Move quickly in vim:
 - #g to move to a specific line
 - gg to move to the top
 - G to move to the bottom

Homework 2 is posted on the website

`/ee/200/public_html/homework/homework_02.txt`

submit it with provide:

```
provide ee200 hw2 <FILE>
```

Bonus material