

```
int main  
return 0;
```

→ OS

```
int f(int x) {    ox free.  
}
```

```
int f(int x) {  
    ...  
}
```

text

first = 8 0b 1000

second = 9 0b 1001

third = 1 0b 0001

third: (first < 4) | (second & first | third)

0b 100000001

0b 1001
& 0b 1000

0b 1000

0b 1000
1 0b 0001

0b 1001

10001001

1 + 8 + 128

137

Anything in the stack gets destroyed

```
char *get_name(void) {  
  char name[128] = ""; char *name = malloc(128);  
  scanf("%127[^\n]", name);  
  return name;  
}
```

#define num 2

printf("%d", num); $\Rightarrow$ printf("%d", 2);

#define ABS(x) ((x < 0) ? (-x) : (x))

```
FILE * fp = fopen("txt", "r");  
char buf[128] = "";
```

```
int * n = malloc(sizeof(int));
```

```
fscanf(fp, "%d %s %x", n, buf, &num2);
```

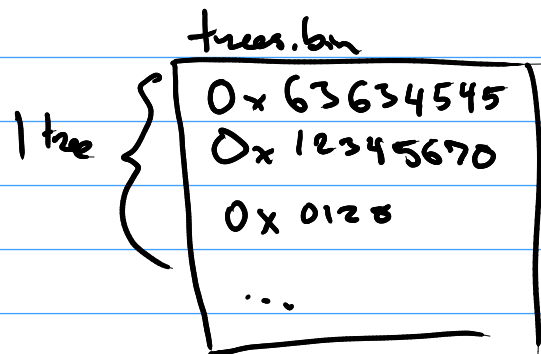
"read all whitespace"

128; word; 64 = "%d; %[%i]; %x"

```
fscanf(fp, "%*d; %[%i]; %*x", buf)
```

binary file I/O

```
struct tree {  
    struct tree *left;  
    struct tree *right;  
    int val;  
}
```



```
FILE *fp = fopen("out", "wb");  
fwrite(tree_buf, sizeof(tree_buf), 1, fp);  
fclose(fp);
```

write 1 tree
to a file

```
FILE *fp = fopen("in", "rb");  
fread(tree_buf, sizeof(tree_buf), 1, fp);  
fclose(fp);
```

read 1 tree
from a file

