

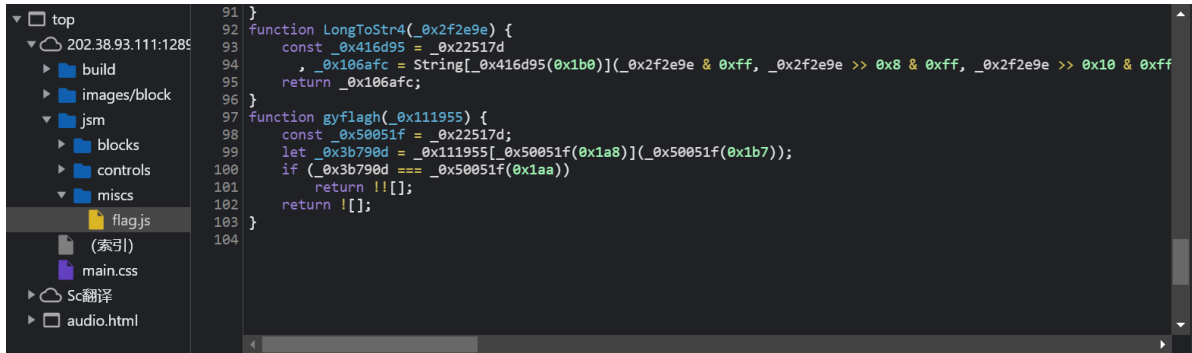


```

if(cinput.length>=32){
    let tbool=gyflagh(cinput.join(''));
    if(tbool) {
        pressplateList[65].TurnOn_redstone_lamp();
        content.innerText='Congratulations!!!';
        return;
    }
}

```

可以看一下段，然后追溯 gyflagh 这个函数,发现他在 flag.js 这个文件里。



初步看一眼，文件 js 混淆了。目前还没有能够有效反混淆的工具，但是我们可以选择调试。第一种，直接浏览器调试，但是被出题人 debug 了。第二中就是本地 nodejs 调。

```

function gyflagh(_0x111955) {
    const _0x50051f = _0x22517d;
    console.log(_0x111955);
    let _0x3b790d = _0x111955[_0x50051f(0x1a8)](_0x50051f(0x1b7));
    console.log(_0x3b790d);
    console.log(_0x50051f(0x1aa));
    if (_0x3b790d === _0x50051f(0x1aa)) return !![];
    return !![];
}

```

主要观察一下这个函数，我们发现只要最后一个if语句成立，我们就可以获得flag。那不妨输出一下  
\_0x50051f(0x1aa)) 发现是

```
6fbde674819a59bfa12092565b4ca2a7a11dc670c678681daf4afb6704b82f0c
```

这时结合这个题目，发现根本没有输出flag的这一功能的函数，所以题目的大概意思是输入flag然后验证。而这里一大长串的就是加密后的flag。所以思路就是读懂其中的加密算法。过程确实有点痛苦，但是只要一点一点调是可以的。

逐步调试后发现code()这个函数就是加密函数。我们只需将其反过来就行了。好像也不难+=改成-=，顺序倒一下就好了。

```

function decode(_0x167a71, _0x762113) {
    let _0x412874 = _0x167a71[0x0],
        _0x3f9c14 = _0x167a71[0x1];
    const _0x540f95 = (0x52cfb2de + 0x4b67c6db),
        _0x2bdc23 = _0x540f95 * 0x20;
    let _0x4f8e47 = _0x2bdc23;
    while (_0x4f8e47 != 0x0) {
        _0x3f9c14 -= (_0x412874 << 0x4 ^ _0x412874 >>> 0x5) + _0x412874 ^
        _0x4f8e47 + _0x762113[_0x4f8e47 >>> 0xb & 0x3];
        _0x4f8e47 -= _0x540f95;
    }
}

```

```

        _0x412874 -= (_0x3f9c14 << 0x4 ^ _0x3f9c14 >>> 0x5) + _0x3f9c14 ^
        _0x4f8e47 + _0x762113[_0x4f8e47 & 0x3];
    }
    _0x167a71[0x0] = _0x412874, _0x167a71[0x1] = _0x3f9c14;
    console.log(_0x167a71[0x0], _0x167a71[0x1]);
}

```

```

104 console.log(Base16ToLong("819a59bf"));
105 console.log(Base16ToLong("a1209256"));
106 console.log(Base16ToLong("5b4ca2a7"));
107 console.log(Base16ToLong("a11dc670"));
108 console.log(Base16ToLong("c678681d"));
109 console.log(Base16ToLong("af4afb67"));
110 console.log(Base16ToLong("04b82f0c"));
111
112 console.log(LongToStr4("10290160461"), LongToStr4("-6991555998"));
113 console.log(LongToStr4("1817013865"), LongToStr4("-3196959890"));
114 console.log(LongToStr4("-3433073615"), LongToStr4("5896746337"));
115 console.log(LongToStr4("14852645681"), LongToStr4("-6824562577"));

```

问题 输出 终端 调试控制台

Code

```

-1354040473
79179532
McWe bRE_
inMl nCrA
1t_3 a5y_
1cIu op9i

```

反正最后flag就这样出了，转码稍微注意一下就行了。