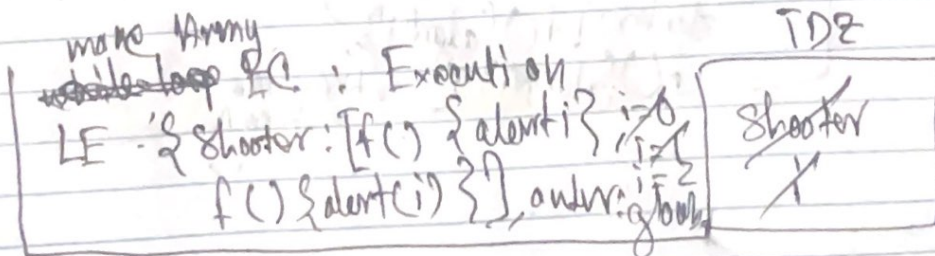
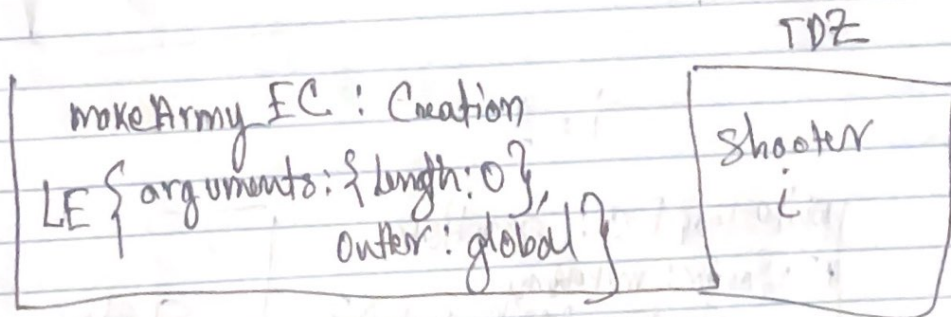
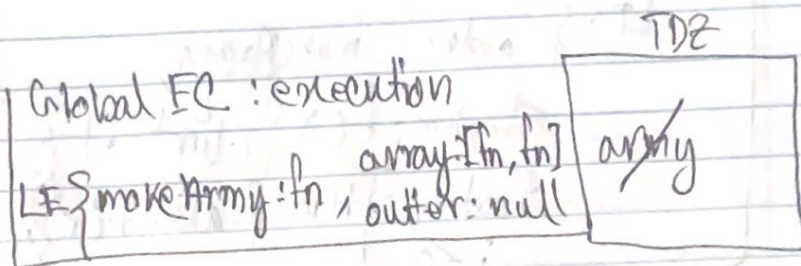
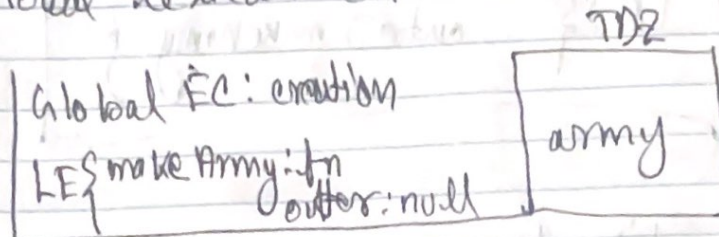


Assignment 6

Q1)

Global Lexical environment (LE)



$i = 0$

while loop EC: execution
LF: { outer: makeArmy }

TDE-

Shooter

while loop EC: execution

LF: { outer: makeArmy
shooter: f() { alert(i) }
shooter: [fn { alert(i) }]
 $i \neq 0$ }

~~Shooter~~

$i = 1$

while loop EC: execution

LF: { outer: makeArmy
shooter: f() { alert(i) }
shooter: [f() { alert(i) }]
outer: makeArmy }

Shooter

LE for `army(0)`

`i = 2` `outer: army`

`army[0]()` EC: creation

LE: { arguments: { length: 0 }, outer: closure scope }

`army[0]()` EC: execution

LE: { argument: { length: 0 }, outer: closure scope }
`alert(i);`
`i = 2`

`army[0]` alert will be 2.

To ensure alert as 0, define scope variable in while

Need to execute `army[0]()` in that case it will work and alert will become 2.

Due to this there will be change in LE of while, whose ~~value~~ will be 1 and `x` is a local variable.

New Code

```
function makeArmy() {
```

```
  let shooter = [];
```

```
  let i = 0;
```

```
  while (i < 2) {
```

```
    let n = i;
```

```
    let shooter = function() {
```

```
      alert(n);
```

```
    };
```

```
    shooters.push(shooter);
```

```
    i++;
```

```
  }
```

```
}
```

```
return n shooters;
```

```
let army = makeArmy();
```

```
army[0]();
```