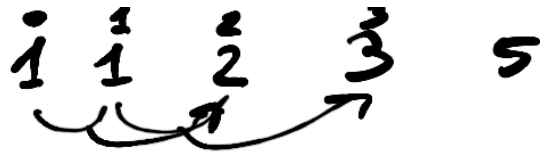
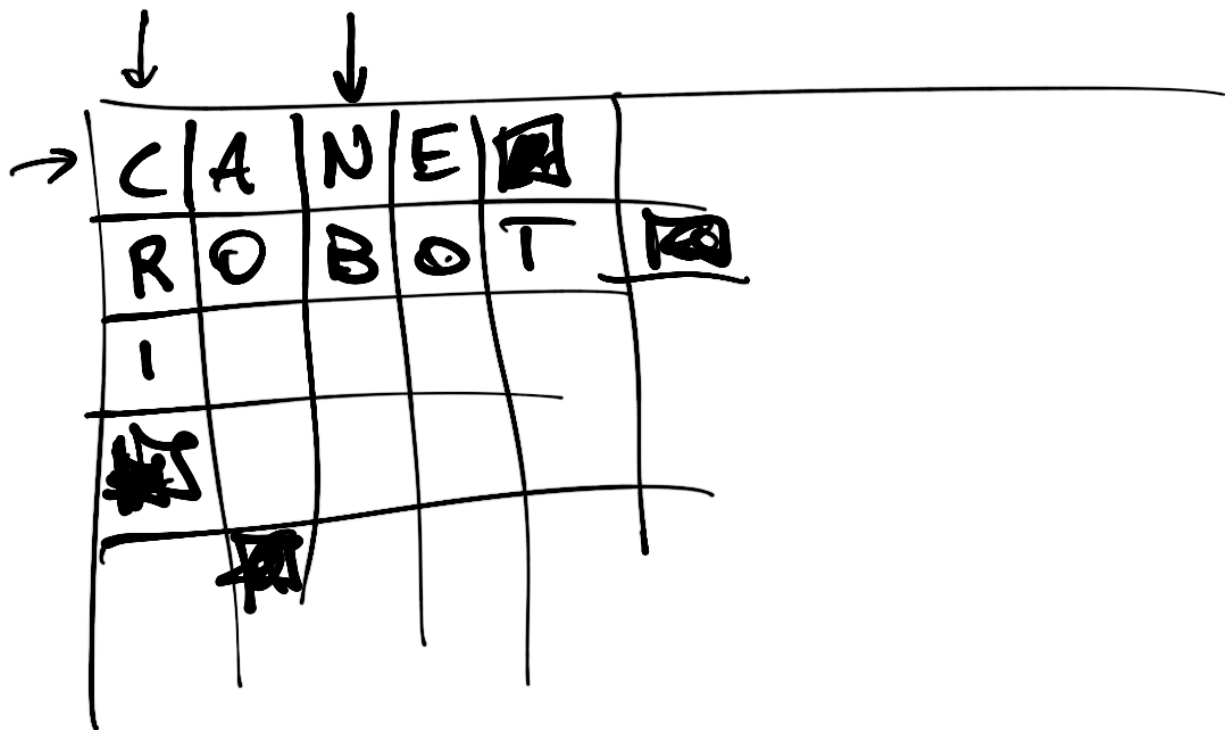


RICORSIONE



```
func fib (n int) int {  
    if n == 0 || n == 1 {  
        return 1  
    }  
    return fib(n-1) + fib(n-2)  
}
```



ANAGRAMMI

cane

canna

ance

acne

ecan

:

ceia

nace

$$\text{anag}("c") = \{ "c" \}$$

$$\text{anag}("cne")$$

$$c \longrightarrow \text{anag}("cne")$$

$$n \longrightarrow \text{anag}("cne")$$

$$e \longrightarrow \text{anag}("cne")$$

$$c \longrightarrow \text{anag}("can")$$

```
func anag (s string) []string {  
if len(s) <= 1 {  
    return []string{s}
```

```
    }  
    var res []string  
    for i := 0; i < len(s); i++ {  
        first := s[i]  
        rest := s[:i] + s[i+1:]  
        anagrams := anag(rest)  
        for _, x := range anagrams {  
            res = append(res,  
                string(first) + x)  
        }  
    }
```

```
    }  
    return res
```

```
}
```

INSIEME DI T



MAPPA

map[T] bool

(si mappa
sempre in
true)

```
func anag (s string) map[string] bool {  
if len(s) <= 1 {  
return map[string] bool {s: true}}
```

```
res := map[string] bool {}  
for i := 0; i < len(s); i++ {  
    first := s[i]  
    rest := s[:i] + s[i+1:]  
    anagrams := anag(s)  
    for x, _ := range anagrams {  
        y := string(first) + x  
        res[y] = true  
    }  
}
```

```
}  
return res
```

```
}
```