

$$m = 11001101111010011100 \quad (2)$$

$$m \ll 1 = 100110111010011000 \quad (2)$$

$$M_d(m \leq 1) = 0100011001110000011000(2)$$

```
m = int(input("m = "))
```

$$l_{\max} = 0$$

while m:

$$l_{\max} = l_{\max} + 1$$
$$m = (m \gg 1) \& m$$

```
print (Quax)
```

④ $A = \{1, 2, \dots, m\}$

$$m = 3$$

$$m=3$$
$$x = (1) \wedge (2) \wedge (3) \wedge (1 \wedge 2) \wedge (1 \wedge 3) \wedge (2 \wedge 3) \wedge (1 \wedge 2 \wedge 3) = 0$$

forță lădă: $0,2 \text{ m}$

Amplas între $\frac{1}{m}$ parte de nr. par de ori în expr. cerută

$$m=0$$
$$m=2; \quad \emptyset; \{1,3\}$$
$$m=2; \emptyset, \{1, 3\}, \{2\}, \{1, 2\}$$
$$m = 3; \emptyset, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}, \{3\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}$$

Elem. m. va fi adăugat la fiecare succesiune de $m-1$ elem.


$$2^{m-1} \rightarrow \text{rec. par}$$

① $S = "$ e a¹b²c b³b⁴a⁵b⁶c a⁷b⁸c a⁹b $"$

$s \leftarrow \text{input} ("s: \dots")$

$t = \text{input} ("t: \dots")$

050

• while true:

 ~~$x = g \cdot \text{rand}(t)$~~

break

y-1

```
print .(p)
```

$$\rho = \rho + \text{len}(t)$$
$$y = 0$$

```
print ("No existe")
```

$s = "aaaaaa"$ 3 parts (non-overlapping = p. distinct)
 $p = p + \text{len}(u)$
 6 or \rightarrow 6 p. superpose (overlapping) $p = p + 1$

② $w = \text{"mare"}$

$\rightarrow p = 2$

"perce", "teste", "programare"

$$m = 3$$

$s = \text{"beste"}$

$p = 10$

$$s[-p:] = \text{"reste"}$$

$\overline{A} = \text{input } (\overline{A} = \overline{\overline{A}})$

`p = int(input("p = "))`

$w = \text{input} ("w = .. ")$

$$x = w [r:]$$
$$e = 11$$

for i in range(m)

```
s = input("s = ")
```

if $s[-p:] = -x$:

$$C_t = S + "L"$$

∴ $c = 1$ ✓

peint ("Nu simțea")

che:

print(c)

③ `s = input("s= ")`

for prim "?!...":

$S = S_{\text{replace}}(p, "u")$

$$l_{\max} = 0$$

гн мхр = " "

```
- for cur in s.split():
```

```
    if len(cur) > lmax:
```

```
        lmax = len(cur)
```

```
        srmex = cur + " "
```

```
elif len(cur) == lmax:
```

```
    if cur not in srmex:
```

```
        srmex = srmex + cur
```

```
print(srmex)
```

④

```
s = "abcabc" | => s = t * 2
```

```
t = "abc" + "abc"
```

if $t \neq s \Rightarrow 1 \leq \text{len}(t) \leq \text{len}(s)/2$

$\text{len}(s) \% \text{len}(t) = 0$

ente prefix of len s

$t * (\text{len}(s) // \text{len}(t)) = s$

```
s = input("s = ")
```

```
for i in range(1, len(s)//2 + 1):
```

```
    if len(s) % i == 0:
```

```
        t = s[:i]
```

```
        k = len(s) // len(t)
```

```
        if t * k == s:
```

```
            print(t, k)
```

```
            break
```

```
else
```

```
    print("we are relative")
```

SEMINAR 2

Șiruri de caractere

1. Să se verifice dacă un șir de caractere t apare ca subșir într-un șir s , iar în caz afirmativ să se afișeze toate pozițiile la care începe t în s . De exemplu, șirul $t = "abc"$ apare ca subșir în șirul $s = "\underline{a}b\underline{c}c\underline{a}b\underline{c}a\underline{b}a\underline{b}c\underline{c}"$ începând cu pozițiile 0, 4 și 9.
2. Se citește un cuvânt w , un număr natural nenul p și un șir format din n cuvinte. Să se afișeze toate cuvintele care sunt p -rime cu w , respectiv ultimele p caractere ale sale coincid cu ultimele p caractere ale lui w . De exemplu, pentru $w = "mere"$, $p = 2$ și cuvintele "pere", "teste" și "programare" (deci $n = 3$), trebuie să fie afișate cuvintele "pere" și "programare".
3. Se citește o propoziție sau o frază în care cuvintele sunt despărțite între ele prin spații și semnele de punctuație uzuale. Să se afișeze toate cuvintele distincte de lungime maximă din propoziția sau fraza dată. De exemplu, în propoziția "Ana are prune și gutui verzi, dar mai multe prune decât gutui!" cuvintele distincte de lungime maximă, egală cu 5, sunt: "prune", "gutui", "verzi", "multe" și "decât".
4. Se citește un șir de caractere s . Să se verifice dacă există un șir t , diferit de s , astfel încât șirul s să se poată obține prin concatenarea de un număr arbitrar de ori k a șirului t . De exemplu, pentru $s = "abcabc"$ avem $t = "abc"$ și $k = 2$.
5. Într-o propoziție, Ana a sintetizat informațiile despre cumpărăturile pe care le-a efectuat într-o anumită zi, pentru fiecare produs cumpărat ea precizând cantitatea și prețul unitar (e.g., "Astăzi am cumpărat 5 kg de mere cu 2.30 RON kilogramul și 2 pâini a câte 5 lei bucata."). Afișați totalul cheltuielilor Anei din ziua respectivă.