

# Alopa !

• bac 2019 :

ex 1 :

Fonction  $\text{whot}(a, b : \text{réel}) : \text{réel}$

Debut

si  $(a - b) \geq 0$  alors

retourner  $a$

sinon

retourner  $\text{whot}(b, a)$

Fin si

Fin

- ①  $\rightarrow$  réel
- ②  $\rightarrow (a - b) \geq 0$
- ③  $\rightarrow 12$
- ④  $\rightarrow c$

ex 2 :

$$f(x) = \frac{1}{x}, x \in ]0; +\infty[ : / \int_1^a f(x) dx :$$

1) fonction surface  $(a : \text{réel}, m : \text{entier}) : \text{réel}$  // Rectangle à gauche :

Debut

$s \leftarrow 0$

$x \leftarrow 1$

$h \leftarrow (a - 1) / m$

Pour i de 1 à m faire

$s \leftarrow s + 1/x$

$x \leftarrow x + h$

fin pour

retourner  $s \times h$

Fin

TDD		
s	réel	surface $\int_1^a f(x) dx$
x	réel	val d'abscisse
h	réel	Longeur
i	entier	compteur

2) fonction Cdcu (a: réel, m: entier): réel.

Debut

$\epsilon \leftarrow 0,0001$

$e \leftarrow a$

tant que  $\text{abs}(\text{surface}(a, m) - 1) > \epsilon$  faire

$e \leftarrow e + \epsilon$

Fin tant que

retourner e

Fin

to check!

Var	type
e	réel
eps	réel
surface	fonction

esc 3

Fonction pgcd (a, b: entier): entier

Debut-

$n \leftarrow a \bmod b$

si  $n = 0$  alors

retourner b

fin si

retourner pgcd (b, n)

Fin

Var	type
n	entier

procedure Inn():

Debut

ouvrir (source, "Fonction.dot", "wb")

ouvrir (Res, "Inn.dot", "wb")

tant que non fin.fichiers (source)

lire (source, objet)

si pgcd (objet.Num, objet.Denom) = 1 alors

crire (Res, objet)

Fin si

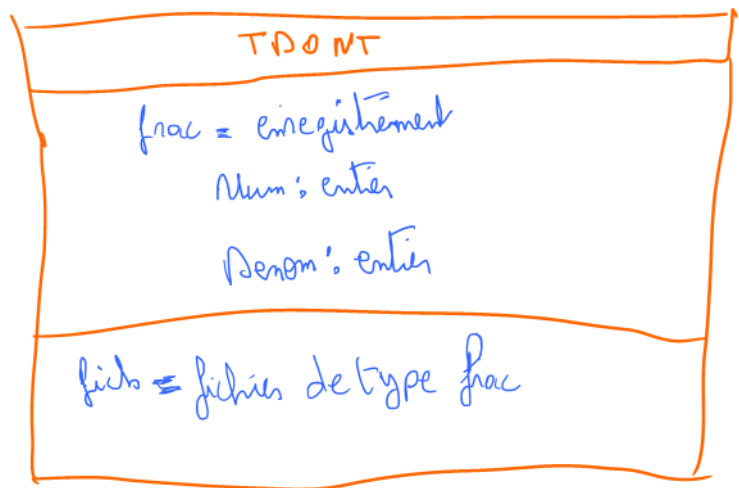
TDOL

Var	type
source	file
Res	file
objet	file
pgcd	fonction

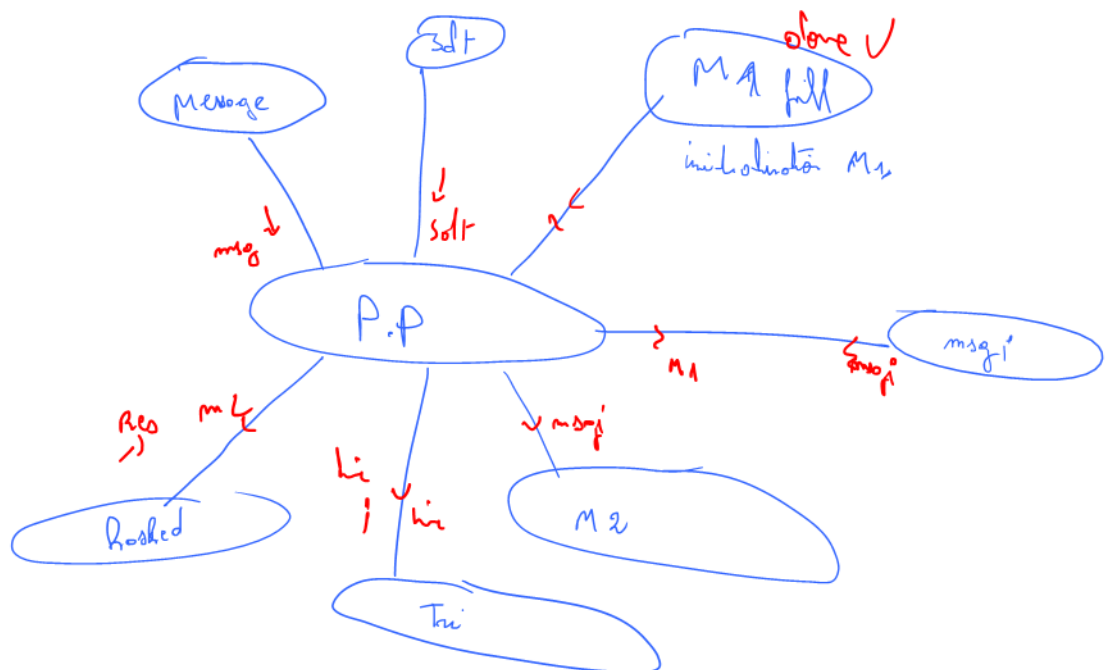
fermer (res)

fermer (source)

Fin



probleme



## Algorithme Hash

Debut

Message  $\leftarrow$  msg()

salt  $\leftarrow$  key()

initialisation( $M_1$ )

Message2  $\leftarrow$  msgi( $M_1$ )

FillM( $M_2$ , Message2, salt)

sort( $M_2[0]$ )

Res  $\leftarrow$  hashed( $M_2$ )

Fin

TDOG	
objet	type
Message, Message2, Salt, Res $M_1$ $M_2$	chaîne de caractères mot 1 mot 2
msg, key, msgi, hashed initialisation, FillM, sort,	Fonction procédure

TDOMT R
$M_1$ : tableau $6 \times 6$ de caractères $M_2$ : tableau $6 \times 7$ de caractères

## module Saisir de messages

Fonction msg() : chaîne

Debut

Repete

good  $\leftarrow$  vrai

lire(ms)

pour i de 0 à Long(ms) - 1 faire

si ord(ms[i])  $\notin$  [97, 122] alors

good  $\leftarrow$  faux

fin si

good  $\leftarrow$  Long[ms]  $\leq$  18

jusqu'à good

Retourner ms

Fin

TDOL	
good ms	boolean chaîne

Fonction `sort()` : chaîne de caractère :

Debut

`good` ← vrai

Repeten

`line(s)`

pour `i` de 0 à `long(s)` - 1 faire