

\* TYPES \*

STR: une suite de quelconque de caractères UTF-8 sans point-virgule, taille bornée seulement par la mémoire disponible

INT: entier relatif compris entre (-sys.maxsize -1) et (sys.maxsize)

FLOAT: nombre à virgule flottante compris entre (sys.float\_info.min) et (sys.float\_info.max)

FLOAT2: nombre à virgule flottante compris entre (sys.float\_info.min) et (sys.float\_info.max), en notation décimale – ex: 123.4567 – arrondi à 8 chiffres après la virgule

\* EX1 \*

INPUT:FLOAT<temps>;FLOAT<distance>

OUTPUT:FLOAT2

\* EX2 \*

INPUT:FLOAT

OUTPUT:FLOAT2

\* EX3 \*

INPUT:STR<mot1>;STR<mot2>

OUTPUT:STR

\* EX4 \*

INPUT:FLOAT<pression>;FLOAT<volume>

OUTPUT:{"K0","Augmenter","Diminuer","OK"}

\* EX5 \*

INPUT:NONE

OUTPUT:[INT, INT, ...]

\* EX6 \*

INPUT:FLOAT<rayon>

OUTPUT:FLOAT2

\* EX7 \*

INPUT:STR<fonction>;INT<borneInf>;INT<borneSup>;INT<nbPas>

OUTPUT:[INT, INT, ...]

\* EX8 \*

INPUT:NONE

OUTPUT:

INT<vitesse>km/h -> INT<heures>;INT<minutes>

INT<vitesse>km/h -> INT<heures>;INT<minutes>

INT<vitesse>km/h -> INT<heures>;INT<minutes>

...

Exemple:

100km/h -> 10:42

110km/h -> 10:32

120km/h -> 10:25

...

\* EX9 \*

INPUT:NONE

OUTPUT:

[INT, INT, ...]

[INT, INT, ...]

[INT, INT, ...]

INT

```
[INT, INT, ...]  
[INT, INT, ...]  
[INT, INT, ...]  
[INT, INT, ...]  
[INT, INT, ...]  
INT
```

```
* EX10 *  
INPUT:NONE  
OUTPUT:[INT, INT, ...]
```

```
* EX11 *  
INPUT:NONE  
OUTPUT:['STR', 'STR', ...]
```

```
* EX12 *  
INPUT:NONE  
OUTPUT:  
X : {'STR', 'STR', ...}  
Y : {'STR', 'STR', ...}  
c in X : True/False  
a in Y : True/False  
X - Y : {'STR', 'STR', ...}  
Y - X : {'STR', 'STR', ...}  
X union Y : {'STR', 'STR', ...}  
X inter Y : {'STR', 'STR', ...}
```

```
* EX13 *  
INPUT:STR  
OUTPUT: {'STR': INT, 'STR': INT, ...} --> trié par ordre alphabétique sur les clés (STR)
```

```
* EX14 *  
INPUT:NONE  
OUTPUT:(INT, INT)
```

```
* EX15 *  
INPUT:FLOAT<x>;FLOAT<y>  
OUTPUT:  
par défaut : (x=FLOAT2, y=FLOAT2)  
instance : (x=FLOAT2, y=FLOAT2)
```

```
* EX16 *  
INPUT:FLOAT<x1>;FLOAT<y1>,FLOAT<x2>;FLOAT<y2>  
OUTPUT:  
v1 : (x=FLOAT2, y=FLOAT2)  
v2 : (x=FLOAT2, y=FLOAT2)  
somme : (x=FLOAT2, y=FLOAT2)
```