

# Programmation C

## Projet : exécution pas à pas de programmes YoctoForth

Alain CROUZIL

`alain.crouzil@irit.fr`

Département d'Informatique  
Faculté Sciences et Ingénierie  
Université Toulouse III – Paul Sabatier

Licence  
Semestre impair 2022-2023

D[i] | Département  
Informatique



# Sommaire

- 1 Introduction
- 2 Mots de contrôle
- 3 Définition et utilisation de nouveaux mots

# Sommaire

- 1 Introduction
- 2 Mots de contrôle
- 3 Définition et utilisation de nouveaux mots

# Introduction

- Les pages qui suivent montrent l'évolution temporelle de la pile de données au fur et à mesure de l'exécution de programmes YoctoForth.
- Pour visualiser cette évolution :
  - utilisez un outil pour afficher les PDF tel qu'Acrobat Reader, Aperçu ou Skim ;
  - affichez le document en mode plein écran ;
  - faites défiler le document page par page.
- Le mot qui vient d'être exécuté est en jaune sur fond rouge.

# Sommaire

- 1 Introduction
- 2 Mots de contrôle**
- 3 Définition et utilisation de nouveaux mots

# Exemple de sélection

## Programme YoctoForth

```
1_2_=_if_1_else_2_endif_.
```

## Description

## Pile de données


## Affichage

# Exemple de sélection

## Programme YoctoForth

```
1 _2_=_if_1_else_2_endif_.
```

## Description

1 est empilé

## Pile de données

1

## Affichage

# Exemple de sélection

## Programme YoctoForh

```
1_2_=_if_1_else_2_endif_.
```

## Description

2 est empilé

## Pile de données

2
1

## Affichage



# Exemple de sélection

## Programme YoctoForh

```
1_2_ if_1_else_2_endif_.
```

## Description

2 et 1 sont dépilés

Le test est faux donc 0 est empilé

## Pile de données

0

## Affichage

# Exemple de sélection

## Programme YoctoForh

```
1_2_=_if_1_else_2_endif_.
```

## Description

0 est dépilé

On passe à `else`

## Pile de données


## Affichage

# Exemple de sélection

## Programme YoctoForh

```
1_2_=_if_1_else_2_endif_. _
```

## Description

## Pile de données


## Affichage

# Exemple de sélection

## Programme YoctoForh

```
1_2_=_if_1_else_2_endif_.
```

## Description

2 est empilé

## Pile de données

2

## Affichage

# Exemple de sélection

## Programme YoctoForh

```
1_2_=_if_1_else_2_endif_._
```

## Description

## Pile de données

2

## Affichage

# Exemple de sélection

## Programme YoctoForh

```
1_2_=_if_1_else_2_endif_
```

## Description

2 est dépilé et affiché

## Pile de données


## Affichage

2

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
while
  __dup__1__+
repeat
.s
```

## Description

## Pile de données


## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1  
begin  
  __dup_4_<  
  while  
    __dup_1_+  
  repeat  
    .s
```

## Description

1 est empilé

## Pile de données

1

## Affichage



# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
while
  __dup__1__+
repeat
.s
```

## Description

## Pile de données

1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup__4__<
  while
    __dup__1__+
  repeat
    .s
```

## Description

1 est empilé

## Pile de données

1
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4__<
  while
    __dup_1__+
  repeat
    .s
```

## Description

4 est empilé


## Pile de données

4
1
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup_4_ 
  while
    __dup_1_+
  repeat
  .s
```

## Description

4 et 1 sont dépilés  
Test vrai donc 1 est empilé

## Pile de données

1
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4_<
  while
  __dup_1_+
  repeat
  .s
```

## Description

1 est dépilé

## Pile de données

1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1  
begin  
  __dup__4__<  
  while  
    __dup__1__+  
  repeat  
  .s
```

## Description

1 est empilé


## Pile de données

1
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1  
begin  
  __dup__4__<  
  while  
    __dup____+  
  repeat  
  .s
```

## Description

1 est empilé

## Pile de données

1
1
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
  while
    __dup__1__
  repeat
  .s
```

## Description

1 et 1 sont dépilés  
2 est empilé

## Pile de données

2
1

## Affichage



# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
  while
    __dup__1__+
  repeat
.s
```

## Description

On passe au mot après `begin`

## Pile de données

2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
while
  __dup__1__+
repeat
.s
```

## Description

2 est empilé

## Pile de données

2
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4__<
  while
    __dup_1__+
  repeat
    .s
```

## Description

4 est empilé


## Pile de données

4
2
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup_4_ 
while
  __dup_1_+
repeat
.s
```

## Description

4 et 2 sont dépilés  
Test vrai donc 1 est empilé

## Pile de données

1
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4_<
  while
  __dup_1_+
  repeat
  .s
```

## Description

1 est dépilé

## Pile de données

2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup__4__<
  while
    __dup__1__+
  repeat
    .s
```

## Description

2 est empilé


## Pile de données

2
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4_<
  while
    __dup_  +
  repeat
    .s
```

## Description

1 est empilé

## Pile de données

1
2
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
  while
    __dup__1__
  repeat
  .s
```

## Description

2 et 1 sont dépilés  
3 est empilé

## Pile de données

3
2
1

## Affichage



# Exemple de répétition

## Programme YoctoForth

```
1  
begin  
  __dup__4__<  
  while  
    __dup__1__+  
    repeat  
  .s
```

## Description

On passe au mot après `begin`

## Pile de données

3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1  
begin  
  __dup__4__<  
  while  
    __dup__1__+  
  repeat  
  .s
```

## Description

3 est empilé

## Pile de données

3
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4__<
while
  __dup_1__+
repeat
.s
```

## Description

4 est empilé


## Pile de données

4
3
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup_4_ 
  while
    __dup_1_+
  repeat
  .s
```

## Description

4 et 3 sont dépilés  
Test vrai donc 1 est empilé

## Pile de données

1
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1  
begin  
  __dup_4_<  
  while  
  __dup_1_+  
  repeat  
  .s
```

## Description

1 est dépilé

## Pile de données

3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup__4__<
  while
    __dup__1__+
  repeat
    .s
```

## Description

3 est empilé


## Pile de données

3
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4_<
  while
    __dup_  +
  repeat
    .s
```

## Description

1 est empilé

## Pile de données

1
3
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
  while
    __dup__1__
  repeat
  .s
```

## Description

3 et 1 sont dépilés  
4 est empilé

## Pile de données

4
3
2
1

## Affichage



# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
  while
    __dup__1__+
    repeat
  .s
```

## Description

On passe au mot après `begin`

## Pile de données

4
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1  
begin  
  __dup__4__<  
while  
  __dup__1__+  
repeat  
  .s
```

## Description

4 est empilé

## Pile de données

4
4
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4__<
while
  __dup_1__+
repeat
.s
```

## Description

4 est empilé


## Pile de données

4
4
4
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup_4_ 
  while
    __dup_1_+
  repeat
  .s
```

## Description

4 et 4 sont dépilés  
Test faux donc 0 est empilé

## Pile de données

0
4
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForth

```
1
begin
  __dup__4__<
  while
  __dup__1__+
  repeat
  .s
```

## Description

0 est dépilé  
On passe après repeat

## Pile de données

4
3
2
1

## Affichage

# Exemple de répétition

## Programme YoctoForh

```
1
begin
  __dup_4_<
  while
    __dup_1_+
  repeat
```



## Description

Le contenu de la pile est affiché

## Pile de données

4
3
2
1

## Affichage

<4>\_1\_2\_3\_4\_

# Sommaire

- 1 Introduction
- 2 Mots de contrôle
- 3 Définition et utilisation de nouveaux mots**

Exemple du mot `somme`

## Programme YoctoForth

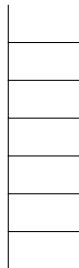
```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

## Pile de données



## Affichage



Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _over _0 _ > _ ( _n > 0 _ ? _
    while
      _over _+ _ ( _ajoute _n _à _la _somme _
      _swap _1 _- _swap _ ( _décrémente _n _
    repeat
      swap _drop _ ( _enlève _n _de _la _pile _
    ;
    4 _somme _ .

```

## Description

4 est empilé

## Pile de données

4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_ somme_.

```

## Description

On passe au premier mot de la définition de `somme`

## Pile de données

4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme _des _n _premiers _entiers _ )
0 _ ( _valeur _initiale _de _la _somme _ )
begin
  _over _0 _ > _ ( _n > 0 _ ? _ )
  while
    _over _+ _ ( _ajoute _n _à _la _somme _ )
    _swap _1 _- _swap _ ( _décrémente _n _ )
  repeat
    swap _drop _ ( _enlève _n _de _la _pile _ )
  ;
  4 _somme _ .

```

## Description

0 est empilé

## Pile de données

0
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme_des_n_premiers_entiers _ )
0 _ ( _valeur_initiale_de_la_somme _ )
begin
  _over_0 _ > _ ( _n > 0 _ ? _ )
  while
    _over_+ _ ( _ajoute_n_à_la_somme _ )
    _swap_1 _ - _swap _ ( _décrémente_n _ )
  repeat
    swap_drop _ ( _enlève_n_de_la_pile _ )
  ;
  4 _somme _ .

```

## Description

## Pile de données

0
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _ _over _ 0 _ > _ ( _n > 0 _ ? _ )
    while
      _ _over _ + _ ( _ajoute _n _à _la _somme _ )
      _ _swap _ 1 _ - _swap _ ( _décrémente _n _ )
    repeat
      swap _drop _ ( _enlève _n _de _la _pile _ )
    ;
  4 _somme _ .

```

## Description

4 est empilé

## Pile de données

4
0
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0 >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
  ;
  4_ somme_.

```

## Description

0 est empilé

## Pile de données


0
4
0
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_  ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_ somme_ .

```

## Description

0 et 4 sont dépilés  
 Test vrai donc 1 est empilé

## Pile de données

1
0
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme_des_n_premiers_entiers _
  0 _ ( _valeur_initiale_de_la_somme _
  begin
    _over_0 _ > _ ( _n > 0 _ ? _
    while
      _over_+ _ ( _ajoute_n_à_la_somme _
      _swap_1 _ - _swap _ ( _décrémente_n _
    repeat
      swap_drop _ ( _enlève_n_de_la_pile _
    ;
    4 _somme _ .

```

## Description

1 est dépilé

## Pile de données

0
4

## Affichage



Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme_des_n_premiers_entiers _ )
0 _ ( _valeur_initiale_de_la_somme _ )
begin
  _over_0 _ > _ ( _n > 0 _ ? _ )
while
  _over_ + _ ( _ajoute_n_à_la_somme _ )
  _swap_1 _ - _swap_ ( _décrémente_n _ )
repeat
  swap_drop _ ( _enlève_n_de_la_pile _ )
;
4 _somme _ .

```

## Description

4 est empilé

## Pile de données

4
0
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ ■ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_ somme_.

```

## Description

4 et 0 sont dépilés  
4 est empilé

## Pile de données

4
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme_des_n_premiers_entiers _
  0 _ ( _valeur_initiale_de_la_somme _
  begin
    _over _ 0 _ > _ ( _n > 0 _ ? _ )
    while
      _over _ + _ ( _ajoute_n_à_la_somme _ )
      _swap _ 1 _ - _swap _ ( _décrémente_n _ )
    repeat
      swap_drop _ ( _enlève_n_de_la_pile _ )
    ;
  4 _somme _ .

```

## Description

4 et 4 sont permutés

## Pile de données

4
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ ■ _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
  ;
  4_ somme_.

```

## Description

1 est empilé

## Pile de données

1
4
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

1 et 4 sont dépilés  
3 est empilé

## Pile de données

3
4

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ -_ swap ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

3 et 4 sont permutés

## Pile de données

4
3

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme _des _n _premiers _entiers _
0 _ ( _valeur _initiale _de _la _somme _
begin
  _over _0 _ > _ ( _n > 0 _ ? _ )
  while
    _over _ + _ ( _ajoute _n _à _la _somme _ )
    _swap _1 _ - _swap _ ( _décrémente _n _ )
  repeat
    swap _drop _ ( _enlève _n _de _la _pile _ )
  ;
  4 _somme _ .

```

## Description

On passe au mot après `begin`

## Pile de données

4
3

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _- _somme [n] _ , _
somme _des _n _premiers _entiers _ )
0 _ ( _valeur _initiale _de _la _somme _ )
begin
  _ _over _ 0 _ > _ ( _n > 0 _ ? _ )
while
  _ _over _ + _ ( _ajoute _n _à _la _somme _ )
  _ _swap _ 1 _- _swap _ ( _décrémente _n _ )
repeat
  swap _drop _ ( _enlève _n _de _la _pile _ )
;
4 _somme _ .

```

## Description

3 est empilé

## Pile de données

3
4
3

## Affichage



Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _over _ 0 _ > _ ( _n > 0 _ ? _ )
    while
      _over _ + _ ( _ajoute _n _à _la _somme _ )
      _swap _ 1 _ - _swap _ ( _décrémente _n _ )
    repeat
      swap _ drop _ ( _enlève _n _de _la _pile _ )
  ;
  4 _somme _ .

```

## Description

0 est empilé

## Pile de données


0
3
4
3

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _over _0 _  _ ( _n > 0 _ ? _
    while
      _over _+ _ ( _ajoute _n _à _la _somme _
      _swap _1 _- _swap _ ( _décrémente _n _
    repeat
      swap _drop _ ( _enlève _n _de _la _pile _
    ;
  4 _somme _ .

```

## Description

0 et 3 sont dépilés  
 Test vrai donc 1 est empilé

## Pile de données

1
4
3

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _over _0 _> _ ( _n > 0 _ ? _
    while
      _over _+ _ ( _ajoute _n _à _la _somme _
      _swap _1 _- _swap _ ( _décrémente _n _
    repeat
      swap _drop _ ( _enlève _n _de _la _pile _
    ;
    4 _somme _ .

```

## Description

1 est dépilé

## Pile de données

4
3

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
  ;
  4_ somme_.

```

## Description

3 est empilé

## Pile de données


3
4
3

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_  ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_somme_ .

```

## Description

3 et 4 sont dépilés  
7 est empilé

## Pile de données

7
3

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_ somme_.

```

## Description

7 et 3 sont permutés

## Pile de données

3
7

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ ■ _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_somme_.

```

## Description

1 est empilé

## Pile de données

1
3
7

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , ,
  somme_des_n_premiers_entiers_
  0_ ( _valeur_initiale_de_la_somme_ )
  begin
    _over_ 0_ >_ ( _n > 0_ ?_ )
    while
      _over_ +_ ( _ajoute_n_ à _la_somme_ )
      _swap_ 1_ _swap_ ( _décrémente_n_ )
    repeat
      swap_drop_ ( _enlève_n_ de _la_pile_ )
    ;
  4_ somme_ .

```

## Description

1 et 3 sont dépilés  
2 est empilé

## Pile de données

2
7

## Affichage



Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _over _0 _ > _ ( _n > 0 _ ? _ )
    while
      _over _ + _ ( _ajoute _n _à _la _somme _ )
      _swap _ 1 _ - _ swap _ ( _décrémente _n _ )
    repeat
      swap _ drop _ ( _enlève _n _de _la _pile _ )
  ;
  4 _somme _ .

```

## Description

2 et 7 sont permutés

## Pile de données

7
2

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme _des _n _premiers _entiers _
0 _ ( _valeur _initiale _de _la _somme _
begin
  _over _0 _ > _ ( _n > 0 _ ? _ )
  while
    _over _ + _ ( _ajoute _n _à _la _somme _ )
    _swap _1 _ - _swap _ ( _décrémente _n _ )
    repeat
    swap _drop _ ( _enlève _n _de _la _pile _ )
  ;
  4 _somme _ .

```

## Description

On passe au mot après `begin`

## Pile de données

7
2

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _- _somme [n] _ , _
somme _des _n _premiers _entiers _
0 _ ( _valeur _initiale _de _la _somme _
begin
  _ _over _ 0 _ > _ ( _n > 0 _ ? _ )
while
  _ _over _ + _ ( _ajoute _n _à _la _somme _ )
  _ _swap _ 1 _- _swap _ ( _décrémente _n _ )
repeat
  swap _drop _ ( _enlève _n _de _la _pile _ )
;
4 _somme _ .

```

## Description

2 est empilé

## Pile de données

2
7
2

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0 >_ ( _n > 0?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4 _somme_.

```

## Description

0 est empilé

## Pile de données


0
2
7
2

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_  ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_ somme_ .

```

## Description

0 et 2 sont dépilés

Test vrai donc 1 est empilé

## Pile de données

1
7
2

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme_des_n_premiers_entiers _
  0 _ ( _valeur_initiale_de_la_somme _
  begin
    _over_0 _ > _ ( _n > 0 _ ? _
    while
      _over_+ _ ( _ajoute_n_à_la_somme _
      _swap_1 _ - _swap _ ( _décrémente_n _
    repeat
      swap_drop _ ( _enlève_n_de_la_pile _
    ;
    4 _somme _ .

```

## Description

1 est dépilé

## Pile de données

7
2

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme_des_n_premiers_entiers _
0 _ ( _valeur_initiale_de_la_somme _
begin
  _over_0_>_ ( _n>0_?_ )
  while
    _over_+ _ ( _ajoute_n_à_la_somme _
    _swap_1_ _swap_ ( _décrémente_n_ )
  repeat
    swap_drop _ ( _enlève_n_de_la_pile _
;
4_somme_ .

```

## Description

2 est empilé

## Pile de données


2
7
2

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_  ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_somme_ .

```

## Description

2 et 7 sont dépilés  
9 est empilé

## Pile de données

9
2

## Affichage



Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
  ;
  4_ somme_ .

```

## Description

9 et 2 sont permutés

## Pile de données

2
9

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ ■ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

1 est empilé

## Pile de données

1
2
9

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

1 et 2 sont dépilés  
1 est empilé

## Pile de données

1
9

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ -_ swap ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_ somme_ .

```

## Description

1 et 9 sont permutés

## Pile de données

9
1

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme_des_n_premiers_entiers _
0 _ ( _valeur_initiale_de_la_somme _
begin
  _over _0 _> _ ( _n>0 _? _
while
  _over _+ _ ( _ajoute_n_à_la_somme _
  _swap _1 _- _swap _ ( _décrémente_n _
  repeat
    swap_drop _ ( _enlève_n_de_la_pile _
;
4 _somme _ .

```

## Description

On passe au mot après `begin`

## Pile de données

9
1

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _- _somme [n] _ , _
somme _des _n _premiers _entiers _ )
0 _ ( _valeur _initiale _de _la _somme _ )
begin
  _ _over _ 0 _ > _ ( _n > 0 _ ? _ )
while
  _ _over _ + _ ( _ajoute _n _à _la _somme _ )
  _ _swap _ 1 _- _swap _ ( _décrémente _n _ )
repeat
  swap _drop _ ( _enlève _n _de _la _pile _ )
;
4 _somme _ .

```

## Description

1 est empilé

## Pile de données

1
9
1

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0 >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_ somme_ .

```

## Description

0 est empilé

## Pile de données


0
1
9
1

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _over _0 _  _ ( _n > 0 _ ? _
    while
      _over _+ _ ( _ajoute _n _à _la _somme _
      _swap _1 _- _swap _ ( _décrémente _n _
    repeat
      swap _drop _ ( _enlève _n _de _la _pile _
    ;
  4 _somme _ .

```

## Description

0 et 1 sont dépilés

Test vrai donc 1 est empilé

## Pile de données

1
9
1

## Affichage



Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
  somme _des _n _premiers _entiers _
  0 _ ( _valeur _initiale _de _la _somme _
  begin
    _over _0 _> _ ( _n > 0 _ ? _
    while
      _over _+ _ ( _ajoute _n _à _la _somme _
      _swap _1 _- _swap _ ( _décrémente _n _
    repeat
      swap _drop _ ( _enlève _n _de _la _pile _
    ;
    4 _somme _ .

```

## Description

1 est dépilé

## Pile de données

9
1

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
  ;
  4_ somme_.

```

## Description

1 est empilé

## Pile de données

1
9
1

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ ■ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

1 et 9 sont dépilés  
10 est empilé

## Pile de données

10
1

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme_des_n_premiers_entiers _ )
0 _ ( _valeur_initiale_de_la_somme _ )
begin
  _over _ 0 _ > _ ( _n > 0 _ ? _ )
  while
    _over _ + _ ( _ajoute_n_à_la_somme _ )
    _swap _ 1 _ - _swap _ ( _décrémente_n _ )
  repeat
    swap_drop _ ( _enlève_n_de_la_pile _ )
  ;
  4 _somme _ .

```

## Description

10 et 1 sont permutés

## Pile de données

1
10

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ ■ -_swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_somme_.

```

## Description

1 est empilé

## Pile de données

1
1
10

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

1 et 1 sont dépilés  
0 est empilé

## Pile de données

0
10

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ -_ swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_.

```

## Description

0 et 10 sont permutés

## Pile de données

10
0

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme _des _n _premiers _entiers _
0 _ ( _valeur _initiale _de _la _somme _
begin
  _over _0 _> _ ( _n > 0 _ ? _ )
  while
    _over _+ _ ( _ajoute _n _à _la _somme _ )
    _swap _1 _- _swap _ ( _décrémente _n _ )
  repeat
    swap _drop _ ( _enlève _n _de _la _pile _ )
  ;
  4 _somme _ .

```

## Description

On passe au mot après `begin`

## Pile de données

10
0

## Affichage



Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_somme_ .

```

## Description

0 est empilé

## Pile de données

0
10
0

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0 _>_ ( _n>0?_ )
  while
    _over_ +_ ( _ajoute_n_à_la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_de_la_pile_ )
;
4_somme_ .

```

## Description

0 est empilé

## Pile de données


0
0
10
0

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_  ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_ somme_ .

```

## Description

0 et 0 sont dépilés  
 Test faux donc 0 est empilé

## Pile de données

0
10
0

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_drop_ ( _enlève_n_ de _la_pile_ )
;
4_somme_ .

```

## Description

0 est dépilé

On passe après `repeat`

## Pile de données

10
0

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme_ ( _n_ - _somme[n]_ , _
somme_des_n_premiers_entiers_ )
0_ ( _valeur_initiale_de_la_somme_ )
begin
  _over_ 0_ >_ ( _n > 0_ ?_ )
  while
    _over_ +_ ( _ajoute_n_ à _la_somme_ )
    _swap_ 1_ - _swap_ ( _décrémente_n_ )
  repeat
    swap_ drop_ ( _enlève_n_ de _la_pile_ )
;
4_ somme_ .

```

## Description

10 et 0 sont permutés

## Pile de données

0
10

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme _des _n _premiers _entiers _ )
0 _ ( _valeur _initiale _de _la _somme _ )
begin
  _over _0 _ > _ ( _n > 0 _ ? _ )
  while
    _over _+ _ ( _ajoute _n _à _la _somme _ )
    _swap _1 _- _swap _ ( _décrémente _n _ )
  repeat
    swap _ drop _ ( _enlève _n _de _la _pile _ )
  ;
  4 _somme _ .

```

## Description

0 est dépilé

## Pile de données

10

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

: _somme _ ( _n _ _somme [n] _ , _
somme _des _n _premiers _entiers _
0 _ ( _valeur _initiale _de _la _somme _
begin
  _over _0 _> _ ( _n > 0 _ ? _ )
  while
    _over _+ _ ( _ajoute _n _à _la _somme _ )
    _swap _1 _- _swap _ ( _décrémente _n _ )
  repeat
    swap _drop _ ( _enlève _n _de _la _pile _ )
  4 _somme _ .

```



## Description

On passe au mot qui suit `somme`

## Pile de données

10

## Affichage

Exemple du mot `somme`

## Programme YoctoForth

```

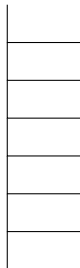
: _somme _ ( _n _ _somme [n] _ , _
  somme_des_n_premiers_entiers _
  0 _ ( _valeur_initiale_de_la_somme _
  begin
    _over_0 _ > _ ( _n > 0 _ ? _
    while
      _over_+ _ ( _ajoute_n_à_la_somme _
      _swap_1 _ - _swap _ ( _décrémente_n _
    repeat
      swap_drop _ ( _enlève_n_de_la_pile _
    ;
  4 _somme _

```

## Description

10 est dépilé et affiché

## Pile de données



## Affichage

10