

IFT 1063: Pseudocode utilisé dans le cours

- **Opérateurs**

:= : opérateur d'affectation

et : opérateur logique \wedge

ou : opérateur logique \vee

non : opérateur logique \neg (négation)

Remarque : Les fonctions et opérateurs déjà définis dans le cours peuvent être utilisés, en plus des opérateurs arithmétiques usuels (par exemple: mod, $\lfloor \rfloor$, $\lceil \rceil$).

- **début**

suite d'instructions

fin

début et **fin** indiquent respectivement le début et la fin de la *suite d'instructions* (facultatif si la *suite d'instructions* n'est composée que d'une seule instruction)

- **si condition alors**

suite d'instructions 1

sinon

suite d'instructions 2

Si la *condition* est vraie, la *suite d'instructions 1* est exécutée, sinon la *suite d'instructions 2* est exécutée (**sinon** est facultatif).

- **tant que condition**

suite d'instructions

instruction suivante

Tant que la *condition* est vraie, on répète la *suite d'instructions* (dès que la *condition* est fausse, on passe à l'*instruction suivante*).

- **pour var := init à limit**

suite d'instructions

instruction suivante

La variable entière *var* est initialisée à la valeur entière *init*; puis, tant que $var \leq limit$ (où *limit* est une valeur entière), on répète la *suite d'instructions* en incrémentant *var* de 1 à chaque répétition (dès que $var > limit$, on passe à l'*instruction suivante*).

- **procédure proc (entrée: e_1, \dots, e_n ; sortie: s_1, \dots, s_m)**

suite d'instructions

Indique l'entête de la procédure *proc*, qui consiste de la *suite d'instructions* qu'on peut exécuter en appelant la procédure ainsi : $proc(e_1, \dots, e_n)$.

Si la *sortie* de la procédure consiste d'une seule valeur, on pourra l'affecter dans une variable du même type : $x := proc(e_1, \dots, e_n)$.

Si la *sortie* de la procédure est une valeur logique, on pourra s'en servir pour tester une condition.