

ALGORITHMIQUE - TI-82 STAT.fr

Les touches indiquent les touches à taper et les AFFICHAGES indiquent les affichages écran.

1 Créer, éditer et exécuter un programme

1.1 Menu programmation

Touche `prgm`, choisir `NOUV` ou `EDIT` ou `EXEC` puis `entrer`.

- `NOUV` permet de créer un nouveau programme :
taper le nom du programme, par exemple `HELLO`, puis `entrer`.
- `EDIT` permet d'écrire et corriger un programme :
choisir le programme, par exemple `1:HELLO`, puis `entrer`.
- `EXEC` permet d'exécuter un programme :
choisir le programme, par exemple `1:HELLO`, puis `entrer`.

1.2 Commandes de programmation

Dans l'éditeur de programme, vous avez accès aux commandes en appuyant sur :

- `prgm` : pour les instructions de programmation (faire défiler avec `▶`) If, For, Input, Disp...
- `math` : pour les fonctions mathématiques ▶Frac, abs, NbrAléat...
- `2nde` + `math` → `tests` : pour les différents tests =, ≠, <, ≤, ≥...
- `2nde` + `mode` → `quitter` : pour quitter le programme

Exemple 1. Un premier programme qui affiche "HELLO WORLD"

```
PROGRAM:HELLO
:Disp "HELLO WORLD"
```

2 Les instructions simples

2.1 Les entrées

La commande `:Input "A=",A` affiche `A=` à l'écran, attend une réponse et stocke cette valeur dans `A`

La commande `:Prompt A` n'affiche rien à l'écran, attend une réponse et stocke cette valeur de `A`

Exemple 2. Deux méthodes pour demander la valeur de A et B

```
:Input "A=",A
:Input "B=",B
```

```
:Prompt A
:Prompt B
```

ou
:Prompt A,B

2.2 Les affectations

Pour affecter, c'est-à-dire stocker une valeur dans une variable, on utilise `sto→` qui affichera `→`

Exemple 3. Dans la variable A, stocker $\frac{1}{4}$ et dans la variable B, stocker le résultat de A^2 :

```
:1/4→A
:A^2→B
```

2.3 Les sorties

La commande `:Disp "A=",A` affiche à l'écran `A=valeur_de_A`

Exemple 4. Affecter 5 à la variable A puis afficher : `A=5` :

```
:5→A
:Disp "A=",A
```

3 La structure de programme

```
PROGRAM:EXEMPLE
:Input "A=",A
:Input "B=",B
:100→N
:A+B→S
:
: bloc_d_instructions_si/pour/tantque;
:
:Disp "W=",W
```

4 Les blocs d'instructions : tests et boucles

4.1 Les instructions conditionnelles (ou tests) « Si » ou « If ».

```
Si ... alors ...
:If condition
:Then
:action
:End
```

```
Si ... alors ... sinon ...
:If condition
:Then
:action1
:Else
:action2
:End
```

Exemple 5. Pour stocker le maximum de A et B dans M :

```
:If A>B
:Then
:A→M
:Else
:B→M
:End
```

4.2 Les boucles « Pour » ou « For »

```
Pour I de A jusque B faire ...
:For(I,A,B)
:action
:End
```

```
Pour I de A jusque B pas P faire ...
:For(I,A,B,P)
:action
:End
```

Exemple 6. Calculer la somme des 100 premiers nombres entiers :

```
:0→S // la somme S est initialisée à 0
:For(I,1,100)
:S+I→S // dans la somme S, on stocke S augmenté de I (I variant de 1 à 100)
:End
:Disp S
```

4.3 Les boucles « Tant que » ou « While »

```
Tant que ... faire ...
:While condition
:action
:End
```

Exemple 7. Lancer un dé tant que l'on obtient pas 6 et compter le nombre de lancers nécessaire :

```
:0→D // le dé D est initialisé à 0
:0→C // le compteur C est initialisé à 0
:While D<6
:ent(NbrAléat*6)+1→D // dans le dé D, on stocke un nombre entier aléatoire entre 1 et 6
:C+1→C // dans le compteur C, on stocke C augmenté de 1
:End
:Disp C
```

Tableaux récapitulatifs des commandes

affectation	:a→2	dans la variable a, on stocke le nombre 2
entrée expression	:Input "A=",A	afficher « A= » à l'écran et stocker la réponse dans A
sortie	:Disp "A=",A	afficher « la valeur de a » à l'écran afficher « a=la valeur de a » à l'écran
arrêt dans une boucle		arrête la boucle
test/alternative	:If condition :Then :action :End	:If condition :Then :action1 :Else :action2 :End
boucle pour	:For(I,A,B) :action :End	:For(I,A,B,P) :action :End
boucle tantque	:While condition :action :End	

Exemples de programmes TI

Exemple 8. Calcul de la **distance** entre deux points $A(x_A; y_A)$ et $B(x_B; y_B)$.

```
:Input "xA=",E
:Input "yA=",F
:Input "xB=",S
:Input "yB=",T
: $\sqrt{(S-E)^2+(T-F)^2}$ →D
:Disp D
```

Exemple 9. Calcul du **coefficient directeur** et de l'**ordonnée à l'origine** de la droite passant par les points $A(x_A; y_A)$ et $B(x_B; y_B)$.

```
:Input "xA=",E
:Input "yA=",F
:Input "xB=",S
:Input "yB=",T
:(T-F)/(S-F)→A
:T-A*S→B
:EffEcr
:Disp "Y=AX+B"
:Disp "A=",A
:Disp "B=",B
```

Exemple 10. Calcul des **solutions de l'équation du second degré** $ax^2 + bx + c = 0$.

```
:Input "A=",A
:Input "B=",B
:Input "C=",C
:B^2-4*A*C→D
:"DELTA=",B
:If D>0
:Then
:Disp "X1=",(-B- $\sqrt{D}$ )/(2*A)►Frac
:Disp "X2=",(-B+ $\sqrt{D}$ )/(2*A)►Frac
:Else
:If D=0
:Disp "X0=", -B/(2*A)►Frac
:Else
:Disp "PAS DE SOLUTION"
:End
:End
```

Exemple 11. On lance deux dés à 4 faces **N** fois de suite. On s'intéresse au pourcentage de fois où l'on obtient une somme égale à **S**.

```
PROGRAM:DE4
:Input "S=",S
:Input "N=",N
:0→C
:For(I,1,N)
:entAléat(1,4)+entAléat(1,4)→R
:If R=S
:Then
:C+1→C
:End
:End
:Disp S,C*100/N
```

Exemple 11. On lance deux dés à 4 faces **N** fois de suite. On s'intéresse au pourcentage de fois où l'on obtient une somme égale à **S**.

```
PROGRAM:DE4
:Input "S=",S
:Input "N=",N
:0→C
:For(I,1,N)
:entAléat(1,4)+entAléat(1,4)→R
:If R=S
:Then
:C+1→C
:End
:End
:Disp S,C*100/N
```

Exemple 11. On lance deux dés à 4 faces **N** fois de suite. On s'intéresse au pourcentage de fois où l'on obtient une somme égale à **S**.

```
PROGRAM:DE4
:Input "S=",S
:Input "N=",N
:0→C
:For(I,1,N)
:entAléat(1,4)+entAléat(1,4)→R
:If R=S
:Then
:C+1→C
:End
:End
:Disp S,C*100/N
```

Exemple 11. On lance deux dés à 4 faces **N** fois de suite. On s'intéresse au pourcentage de fois où l'on obtient une somme égale à **S**.

```
PROGRAM:DE4
:Input "S=",S
:Input "N=",N
:0→C
:For(I,1,N)
:entAléat(1,4)+entAléat(1,4)→R
:If R=S
:Then
:C+1→C
:End
:End
:Disp S,C*100/N
```

Remplir une liste (équivalent de la commande seq ?) : `listes` `OPT` `Remplir(`

