

# Langages et environnements de programmation

Épreuve pratique de TP ALGO – Concours ENS

## 1 Choix du langage de programmation

Trois principaux langages de programmation principaux vous sont proposés : Python (**filière MP uniquement**), OCaml et C.

## 2 Langage Python (filière MP uniquement)

Par défaut, la version utilisée est Python 3 (version 3.11.2). Mais Python 2 peut également être disponible (version 2.7.18) dans certaines configurations, voir ci-dessous.

### 2.1 Environnement de développement Pyzo

Python 3 est disponible dans l’environnement de développement Pyzo. Pour lancer Pyzo, utiliser “Applications” → “Développement” → “Pyzo”.

### 2.2 Environnement de développement Spyder

Python 3 est disponible dans l’environnement de développement Pyzo. Pour lancer Spyder, utiliser “Applications” → “Développement” → “Spyder”.

### 2.3 Éditeur Gedit avec exécution intégrée (greffon "External tools")

Vous trouverez l’éditeur Gedit dans le menu “Applications” → “Accessoires”.

Pour exécuter votre code avec Python vous pouvez utiliser le raccourci suivant : touche *Majuscule* + touche *F5*. L’exécution se fera dans la partie inférieure de la fenêtre. Le code est automatiquement sauvegardé avant son exécution.

Par défaut, Python 3 est utilisé. Si vous le souhaitez, vous pouvez utiliser Python 2 en configurant le raccourci :

- Cliquez sur le bouton avec les trois barres horizontales, puis sur “Gestion des outils externes”,
- Sélectionnez l’outil “Python” dans le panneau de gauche,
- Dans la 2ème ligne du script, remplacez “python3” par “python2”.

### 2.4 Visual Studio Code

Vous le trouverez dans le menu “Applications” → “Développement” . Attention lors de la sauvegarde de vos fichiers, sélectionner d’abord un emplacement (Dossier personnel ou Bureau).

Pour exécuter votre code Python, vous pouvez lancer Python sur votre fichier (“Run Python File”). Vous pouvez aussi exécuter une sélection de code dans une console Python (combinaison

de touches *Majuscule* + *Entrée*, ou bien clique-droit → “Run Python” → “Run Selection/Line in Python Terminal”).

## 2.5 Jupyter

Jupyter peut être lancé dans le menu “Applications” → “Développement”.

Avec Jupyter Notebook, dans la page qui apparaît dans le navigateur, cliquer sur le bouton “Nouveau” à droite de l’écran et choisir “Python 3 (ipykernel)”.

Avec JupyterLab, vous pouvez au choix utiliser un notebook Jupyter, ou lancer une console Python 3 (ipykernel).

## 2.6 Python 2

Pour passer en Python 2, il faut exécuter dans un terminal la commande `pyenv global 2.7.18`. La commande `python` lancera alors l’interpréteur Python en version 2.7.18.

Pour revenir à Python 3, il faut exécuter dans un terminal la commande `pyenv global system`. La commande `python` lancera alors l’interpréteur Python en version 3.11.2.

# 3 Langage OCaml

Ocaml est disponible en version 5.1.1, sauf avec Jupyter, dans lequel OCaml n’est disponible qu’en version 4.14.2.

## 3.1 Éditeur Gedit avec exécution intégrée (greffon "External tools")

Vous trouverez l’éditeur Gedit dans le menu “Applications” → “Accessoires”.

Pour exécuter votre code avec OCaml, vous pouvez utiliser le raccourci suivant : touche *Majuscule* + touche *F4*. L’exécution se fera dans la partie inférieure de la fenêtre. Le code est automatiquement sauvegardé avant son exécution.

## 3.2 Visual Studio Code

Vous le trouverez dans le menu “Applications” → “Développement” .

En ouvrant un fichier `.ml`, vous pouvez avoir accès aux fonctionnalités offertes par Visual Studio Code (typage, prototypes de fonction, etc.). Vous pouvez exécuter votre code dans un terminal intégré à l’aide de la combinaison de touches *Majuscule* + *Entrée*, ou le compiler avec `ocamlc` ou `ocamlopt` avant de l’exécuter.

## 3.3 Éditeur Emacs avec mode Tuareg

Vous pouvez utiliser l’éditeur Emacs avec le mode Tuareg. Avec cette option, vous éditez votre fichier source, et l’exécuterez à travers l’éditeur.

Pour lancer Emacs, vous pouvez utiliser “Applications” → “Accessoires” → “Emacs (GUI)”, ou “Applications” → “Développement” → “Emacs (GUI)”. Vous pouvez également le lancer depuis un terminal, avec la commande `emacs`.

Emacs est un éditeur très puissant, mais qui peut dérouter en premier lieu, notamment car il n’utilise pas les mêmes raccourcis que ceux standards de Windows. Néanmoins, il est possible

d'utiliser le menu pour toutes les opérations importantes (lire et enregistrer un fichier, copier et coller des zones de texte).

Le mode Tuareg s'active automatiquement si vous ouvrez un fichier avec l'extension `.ml`. (Sinon, pour forcer Emacs à passer en mode Tuareg, appuyez sur *Échap* puis `x`, et tapez `tuareg-mode` puis *Entrée*.)

Deux raccourcis sont importants à connaître :

- “`ctrl-c`” (touche *Control* conjointement avec la touche `c`) puis “`ctrl-b`” : exécute tout le fichier.
- “`ctrl-c`” puis “`ctrl-e`” : la ligne courante.

Lors de la première exécution, il vous demandera quel interpréteur choisir. Par défaut, `ocaml` est utilisé.

### 3.4 Un éditeur quelconque, et exécution dans un terminal

Vous aurez deux fenêtres ; un éditeur, et un terminal pour exécuter votre code. Vous pouvez utiliser l'éditeur de votre choix (par exemple `gedit` ou `emacs`). Pour lancer un terminal : “Applications” → “Émulateur de terminal”.

Pour exécuter un code, enregistrez le fichier, et exécutez le dans le terminal avec la commande `ocaml votrefichier.ml`.

### 3.5 Boucle d'interaction OCaml

Vous pouvez directement lancer OCaml en mode interactif (commande `ocaml` dans un terminal). Il est possible de lancer `ledit ocaml` pour avoir des facilités d'édition (gauche/droite et historique). Vous pouvez combiner cette option avec l'option précédente en utilisant la commande `#use "votrefichier.ml";;` dans l'interface interactive pour lire un fichier extérieur. (Attention à sauvegarder votre code si vous utilisez cette méthode !)

### 3.6 Jupyter

Jupyter peut être lancé dans le menu “Applications” → “Développement”.

Avec Jupyter Notebook, dans la page qui apparaît dans le navigateur, cliquer sur le bouton “New” à droite de l'écran et choisir “OCaml default”.

Avec JupyterLab, vous pouvez au choix utiliser un notebook Jupyter, ou lancer une console OCaml.

**Attention : dans Jupyter, Ocaml est disponible uniquement en version 4.14.2.**

## 4 Langage C

### 4.1 Visual Studio Code

Vous le trouverez dans le menu “Applications” → “Développement” .

Vous pouvez éditer vos fichiers `.c` et compiler vos programmes avec `gcc` dans un terminal intégré.

### 4.2 Un éditeur quelconque, compilation et exécution dans un terminal

Vous pouvez utiliser l'un des éditeurs disponibles (`gedit`, `emacs`, ...) puis compiler votre code à l'aide de `gcc` dans un terminal avant de l'exécuter.

### 4.3 Exemple de compilation et d'exécution dans un terminal

Si le fichier `helloworld.c` est

```
#include <stdio.h>
int main (void)
{
    printf("Hello World!\n");
}
```

alors dans un terminal, on a

```
tpalgo@tpalgo:~$ gcc -o helloworld helloworld.c
tpalgo@tpalgo:~$ ./helloworld
Hello World!
tpalgo@tpalgo:~$
```