

## Sommario: 23 Marzo, 2021

- OCaml: Riferimenti
- OCaml: Top Level Interactive Interpreter (REPL)
- Ocaml: Caratteristiche Essenziali
- OCaml: Una panoramica in REPL

# OCaml: Top Level Interactive Interpreter (REPL)

- **Uso.**

(a-e)....

(f) Prova: Accesso Sessione, Scrittura ed Esecuzione di un programma

Caricamento in REPL ed esecuzione:

```
Marco-Bellias-MacBook-Air:~ marcob$ ocaml
OCaml version 4.08.1

# (* Iniziamo una sessione interattiva in REPL di OCaml *);;
# ;;
# open Printf;;
# let rec gcd x y =
    if x=y then x
    else if x>y then gcd (x-y) y else gcd x (y-x);;
val gcd : int -> int -> int = <fun>
# gcd 3 17;;
- : int = 1
# printf "applichiamo gcd\n    gcd 64 6 = %d\n" (gcd 64 6);;
applichiamo gcd
    gcd 64 6 = 2
- : unit = ()
# (* terminiamo la sessione interattiva in REPL *);;
# #quit;;
Marco-Bellias-MacBook-Air:~ marcob$
```

## ● **Multi-Paradigma.** Funzionale - Imperativo - Object Oriented

- Funzionale
  - Nucleo Originario (Caml))
  - Fondamentale: Integrato da costrutti per gli altri Paradigmi
  - Laboratorio: Useremo solo il nucleo Funzionale
- Funzionale: Caratteristiche Essenziali.
  - Derivato dal Modello del Lambda Calcolo
  - Funzioni come Valori: Funzioni per Definire Calcoli e come Valori dei Calcoli
  - Valori Senza Stato: Non Modificabili (niente Variabili)
  - Tipi: Associati ad ogni entità introdotta nel Programma
  - Strutture di Controllo di Base:
    - Composizione e Applicazione di Funzione
    - Blocco Espressione
  - Dati Strutturati di Base: Tuple e Liste con Structure Sharing
  - Programma:
    - Collezione di Moduli: Definiscono il contesto di valutazione
      - Introducono definizioni di entità diverse: Tipi, Parametri, Funzioni, ...
      - Hanno correlazioni e dipendenze
    - Espressione da valutare