

Programmation Fonctionnelle - Corrigé TP 8

Licence d'Informatique, Université Paris XII

Lancelot PECQUET

Date et heure en français – Invoquée depuis un terminal, la commande **date** donne la date et l'heure.

Par exemple:

```
% date
Mon May 03 10:05:18 EDT 2004
```

Nous allons écrire une horloge en Caml qui écrit la date en français, pour donner:

```
% frdate
Lundi 03 Mai 2004, 10:05:18
```

1 Affichage textuel de la date

Tout d'abord, le makefile:

```
bytecode: frdate.ml
    ocamlc -custom unix.cma -o frdate frdate.ml -cclib -lunix

native: frdate.ml
    ocamlc -custom unix.cmxa -o frdate frdate.ml -cclib -lunix

clean:
    rm -f *.cmi *.cmo frdate
```

... et la solution:

```
let frdate_rec = Unix.localtime(Unix.gettimeofday());;
let frdate_heure = frdate_rec.Unix.tm_hour;;
let frdate_min = frdate_rec.Unix.tm_min;;
let frdate_sec = frdate_rec.Unix.tm_sec;;
let frdate_jour_s = frdate_rec.Unix.tm_wday;;
let frdate_mois = frdate_rec.Unix.tm_mon;;
let frdate_jour_m = frdate_rec.Unix.tm_mday;;
let frdate_annee = frdate_rec.Unix.tm_year + 1900;;
```

```

let traduire_jour = function
  0 -> "Lundi"
| 1 -> "Mardi"
| 2 -> "Mercredi"
| 3 -> "Jeudi"
| 4 -> "Vendredi"
| 5 -> "Samedi"
| 6 -> "Dimanche";;

let traduire_mois = function
  0 -> "Janvier"
| 1 -> "Février"
| 2 -> "Mars"
| 3 -> "Avril"
| 4 -> "Mai"
| 5 -> "Juin"
| 6 -> "Juillet"
| 7 -> "Août"
| 8 -> "Septembre"
| 9 -> "Octobre"
| 10 -> "Novembre"
| 11 -> "Décembre";;

print_string (traduire_jour frdate_jour_s);;
print_string " ";;
print_int frdate_mois;;
print_string " ";;
print_string (traduire_mois frdate_jour_m);;
print_string " ";;
print_int frdate_annee;;
print_string ", ";;
print_int frdate_heure;;
print_string ":";;
print_int frdate_min;;
print_string ":";;
print_int frdate_sec;;
print_string "\n";;

```

2 Affichage graphique de la date

Le programme xclock -digital affiche:

Mon May 3 10 : 05: 18 2004

L'objectif, maintenant, est de faire une version graphique **xfrdate** de **frdate** qui affiche en temps réel l'heure courante en français et sur deux lignes:

Lundi 3 Mai 2004
 10:05:18

Voici le programme principal:

```
let lp_width = 150;;
let lp_height = 50;;

Graphics.open_graph (" " ^ string_of_int(lp_width) ^ "x" ^ string_of_int(lp_height));;
Graphics.set_color Graphics.blue;;
Graphics.clear_graph();;
Graphics.fill_rect 0 0 lp_width lp_height;;
Graphics.set_color Graphics.yellow ;; let lp_days =
  ["Sun";"Mon";"Tue";"Wed";"Thu";"Fri";"Sat" []];;
let lp_months =
  ["Jan";"Feb";"Mar";"Apr";"May";"Jun";"Jul";"Aug";"Sep";"Oct";"Nov";"Dec" []];;
Graphics.set_font("9x15bold") ;;

while true do
  Unix.sleep 1;
  Graphics.set_color Graphics.blue;
  Graphics.fill_rect 0 0 lp_width lp_height;
  Graphics.set_color Graphics.yellow;
  let lp_time = Unix.localtime (Unix.gettimeofday ()) in
    let lp_h0 = string_of_int(lp_time.Unix.tm_hour) and
    lp_m0 = string_of_int(lp_time.Unix.tm_min) and
    lp_s0 = string_of_int(lp_time.Unix.tm_sec) and
    lp_d0 = string_of_int(lp_time.Unix.tm_mday) in
      let lp_time_string = (if String.length lp_h0 = 2 then lp_h0 else "0"^lp_h0) ^ ":" ^
        (if String.length lp_m0 = 2 then lp_m0 else "0"^lp_m0) ^ ":" ^
        (if String.length lp_s0 = 2 then lp_s0 else "0"^lp_s0) and
      lp_day_string = lp_days.(lp_time.Unix.tm_wday) ^ " " ^
        lp_months.(lp_time.Unix.tm_mon) ^ " " ^
        (if String.length lp_d0 = 2 then lp_d0 else "0"^lp_d0) ^ " " ^
        string_of_int(1900+lp_time.Unix.tm_year) in

        let (lp_day_width,lp_text_height) = Graphics.text_size lp_day_string and
        (lp_time_width,_) = Graphics.text_size lp_time_string in

          Graphics.moveto (lp_width/2 - lp_day_width/2)
            (lp_height/2 );
          Graphics.draw_string lp_day_string;
          Graphics.moveto (lp_width/2 - lp_time_width/2)
            (lp_height/2 - lp_text_height);
          Graphics.draw_string lp_time_string;

done;;
```

et le **makefile** correspondant

```
bytecode: xfrdate.ml
  ocamlc -custom graphics.cma unix.cma -o xfrdate xfrdate.ml -cclib -lunix -cclib -lX11

native: xfrdate.ml
```

```
ocamlopt graphics.cmxa unix.cmxa -o xfrdate xfrdate.ml -cclib -lunix -cclib -lX11  
clean:  
rm -f *.cmi *.cmo *.cmx *.o xfrdate
```