

camltotex_main

```
1 (* Caml to LaTeX Caml Listing *)
2 * Copyright (C) 2007 Philippe Wang ( mail@philippewang.info ) *
3 * This program is free software; you can redistribute it and/or *
4 * modify it under the terms of the GNU General Public *
5 * License as published by the Free Software Foundation; either *
6 * version 2 of the License, or (at your option) any later version. *
7 * *
8 * This program is distributed in the hope that it will be useful, *
9 * but WITHOUT ANY WARRANTY; without even the implied warranty of *
10 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU *
11 * General Public License for more details. *
12 * *
13 * You should have received a copy of the GNU General Public *
14 * License along with this program; if not, write to the *
15 * Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, *
16 * Boston, MA 02110-1301, USA. *)
17 (*****
18 (* :: *)
19 (*****
20 (* $Id: camltotex_main.ml,v 1.8 2007/06/23 14:48:36 philippe Exp $ *)
21
22 let _ =
23   print_endline "
24   % --- put these few commented definitions before \begin{document} ---
25   % \usepackage{color}
26   % \usepackage{courier}
27   % \usepackage[T1]{fontenc}
28   % \usepackage{palatino}
29   % \usepackage{courier}
30   % \usepackage{alltt}
31   % \DeclareTextSymbol{\QT}{T1}{39}
32   % \DeclareTextSymbol{\COMMA}{T1}{44}
33   % \DeclareTextSymbol{\COLON}{T1}{58}
34   % \DeclareTextSymbol{\SC}{T1}{59}
35   % \DeclareTextSymbol{\BS}{T1}{92}
36   % \DeclareTextSymbol{\CI}{T1}{94}
37   % \DeclareTextSymbol{\TI}{T1}{126}
38   % \definecolor{navy}{rgb}{0.15, 0.15, 0.45}
39   % \definecolor{myblue}{rgb}{0.25, 0.25, 0.645}
40   % \definecolor{darkred}{rgb}{0.845, 0.125, 0.125}
41   % \definecolor{grey}{rgb}{0.4, 0.4, 0.4}
42   % \definecolor{darkgreen}{rgb}{0.125, 0.845, 0.125}
43   % \definecolor{leaf}{rgb}{0.1, 0.9, 0.1}
44   %
45   % \newcommand{\mlkeywordA}[1]{\mbox{\color{cyan}{#1}}}
46   % \newcommand{\mlkeywordB}[1]{\mbox{\color{navy}{#1}}}
47   % \newcommand{\mlkeyword}[1]{\mbox{\color{red}{#1}}}
48   % \newcommand{\mloperator}[1]{\mbox{\color{darkgreen}{#1}}}
49   % \newcommand{\mlmodulename}[1]{\mbox{\color{navy}{#1}}}
50   % \newcommand{\mlstring}[1]{\mbox{\color{navy}{#1}}}
51   % \newcommand{\mlcomments}[1]{\mbox{\color{grey}{#1}}}
52   % \newcommand{\mlcodeline}[2]{\tiny\sl #1 & \begin{minipage}\
53   [c]{0.8\linewidth}\begin{alltt}\mbox{#2}\end{alltt}\end{minipage}}
54   % -----
55   ";
56   print_endline "\n{\scriptsize\noindent\begin{tabular}{r|l}";
57   Printf.printf "\mlcodeline{%d}" 1;
58   let () =
59     Camltotex_lexer.rows_limit := (try int_of_string Sys.argv.(2) with _ -> max_int)
60   in
```

```

61 | let input =
62 |   if Array.length Sys.argv = 2 || Array.length Sys.argv = 3 then
63 |     open_in Sys.argv.(1)
64 |   else
65 |     stdin
66 | in
67 | try
68 |   let lexbuf = Lexing.from_channel input in
69 |   while true do
70 |     ignore (Camltotex_lexer.token lexbuf)
71 |   done
72 | with Camltotex_lexer.Eof ->
73 |   print_endline "\n\\end{tabular}\\n";
74 |   (try close_in input with _ -> ());
75 |   exit 0
76 |

```

md5sum

```

1 | type 'a t = 'a list
2 |
3 | open String module I=Int32 module J=Int64 let n,v,(<),(=),($),(&),(/),(&*),(@),
4 | (+),(>*),w,m,l,(%)=I.lognot, Sys.argv,I.shift_left,I.shift_right_logical,I.add,I
5 | .logand,I.logor,J.logand,I.logxor,J.add,J.shift_right_logical,sub,Array.make,
6 | Array.length,List.iter let (<>) i x=(i<x)|| (i=(32-x)) and r,f=let f x=I.of_int(Char
7 | .code x) in let c x=Char.chr(I.to_int x) in fun n->let s=create 8 in let a,b=J.
8 | to_int32(n>*32),J.to_int32 n in s.[4]<-c(2551&a);s.[5]<-c(a=8&2551);s.[6]<-c(a=
9 | 16&2551);s.[7]<-c(a=24&2551);s.[0]<-c(2551&b);s.[1]<-c(b=8&2551);s.[2]<-c(b=16&
10 | 2551);s.[3]<-c(b=24&2551);s),(fun s->let a,b,c,d=f s.[0],(f s.[1])<8,(f s.[2])<
11 | 16,(f s.[3])<24 in a|b|c|d) let u s=let x=m 16 01 in for i=0to 15do x.(i)<-f(w
12 | s(i*4)4) done;x and t=let t=m 64 01 in for i=0to 1 t-1do t.(i)<-J.to_int32(J.
13 | of_float(ldexp(abs_float(sin(float(i+1))))32)) done;t and g x=let l=J.
14 | to_int((J.sub 960L(511L&*x))&*511L) in let l=if not(l<>0) then 512 else l in let s=
15 | make((l+1)/8)'000' in s.[0]<-'128';s^(r x) let rec i()=j:=(!j+1) mod 64;!j and j=
16 | ref(-1) and h x b a b c d=j:=--1;let aa,bb,cc,dd= !a,!b,!c,!d in let rec x()=p:=(!p
17 | +1) mod 4;match!p with 0->a,b,c,d|3->b,c,d,a|2->c,d,a,b|_>d,a,b,c and p=ref 0 and
18 | o()=p:= -1 in(o());let k=ref(-1) in let m()=incr k;!k in let f s=let(a,b,c,d),k=x()
19 | ,m() in a:= !b$(!a$(!b&!c)|/(n!b)&!d)$xb.(k)$t.(i())><s) in f%[7;12;17;22;7;
20 | 12;17;22;7;12;17;22;7;12;17;22];o();let f(k,s)=let(a,b,c,d)=x() in a:= !b$(!a$(!
21 | b&!d)|/(c&(n!d))$xb.(k)$t.(i())><s) in f%[1,5;6,9;11,14;0,20;5,5;10,9;15,14;4
22 | ,20;9,5;14,9;3,14;8,20;13,5;2,9;7,14;12,20];o();let f(k,s)=let(a,b,c,d)=x() in a
23 | := !b$(!a$(!b@ !c@ !d)$xb.(k)$t.(i())><s) in f%[5,4;8,11;11,16;14,23;1,4;4,11;7,
24 | 16;10,23;13,4;0,11;3,16;6,23;9,4;12,11;15,16;2,23];o();let f(k,s)=let(a,b,c,d)=x
25 | () in a:= !b$(!a$(!c@(!b|/n!d))$xb.(k)$t.(i())><s) in f%[0,6;7,10;14,15;5,21;12
26 | ,6;3,10;10,15;1,21;8,6;15,10;6,15;13,21;4,6;11,10;2,15;9,21];a:= !a$aa;b:= !b$bb
27 | ;c:= !c$cc;d:= !d$dd) let p l f=(fun a->(Printf.printf"%02lx")%[a<24=24;a=8<24=24
28 | ;a=16<24=24;a=24<24=24])%l;Printf.printf "%s\n" f let z f=let a,b,c,d=ref(* * *)
29 | 0x674523011,ref 0xefcdab891,ref 0x98badcfe1,ref 0x103254761 in let ic,bf=open_in
30 | f,create 64 in let rec e ea=match input ic bf 0 64 with 0->h(u(g ea)) a b c d|64->h
31 | (u bf) a b c d;e(ea++512L)|l->let ea=ea+(J.of_int(8*1)) in let s=(w bf 0 l)^(g ea
32 | ) in let sl=length s in if sl>64 then(h(u(w s 0 64)) a b c d;h(u(w s 64(sl-64))) a
33 | b c d) else h(u s) a b c d in e 0L;p[!a;!b;!c;!d]f;for i=1to l v-1do z v.(i) done
34 |

```