#include <stdio.h>

#include <SDL/SDL.h>

int main(int argc, char \*argv[])

{

int ii=0,jj=0;

SDL\_Surface \*screen;

SDL\_Surface \*image,\*a,\*b,\*c,\*d,\*e,\*f,\*h,\*i,\*j,\*l,\*g,\*m,\*n,\*o,\*p,\*k,\*r,\*s,\*t,\*w,\*u,\*v,\*x1,\*y1,\*z,\*q,\*ok,\*ca,\*cadre,\*go;

SDL\_Rect src, dest,dest1,src1,dest2,dest3,dest4,dest5,dest6,dest11,dest22,dest7,dest8,dest9,dest10,dest12;

SDL\_Event event ;

SDL\_EnableUNICODE(1);

if (SDL\_Init(SDL\_INIT\_VIDEO) != 0)

{

printf("Unable to initialize SDL: %s\n", SDL\_GetError());

return 1;

}

screen = SDL\_SetVideoMode(690,499, 32, SDL\_HWSURFACE | SDL\_DOUBLEBUF);

if (screen == NULL)

{

printf("Unable to set video mode: %s\n", SDL\_GetError());

return 1;

}

image = SDL\_LoadBMP("img/11.bmp");

a=SDL\_LoadBMP("img/a.bmp");

b=SDL\_LoadBMP("img/b.bmp");

c=SDL\_LoadBMP("img/c.bmp");

d=SDL\_LoadBMP("img/d.bmp");

e=SDL\_LoadBMP("img/e.bmp");

f=SDL\_LoadBMP("img/f.bmp");

g=SDL\_LoadBMP("img/g.bmp");

h=SDL\_LoadBMP("img/h.bmp");

i=SDL\_LoadBMP("img/i.bmp");

j=SDL\_LoadBMP("img/j.bmp");

k=SDL\_LoadBMP("img/k.bmp");

l=SDL\_LoadBMP("img/l.bmp");

m=SDL\_LoadBMP("img/m.bmp");

n=SDL\_LoadBMP("img/n.bmp");

o=SDL\_LoadBMP("img/o.bmp");

p=SDL\_LoadBMP("img/p.bmp");

k=SDL\_LoadBMP("img/k.bmp");

r=SDL\_LoadBMP("img/r.bmp");

s=SDL\_LoadBMP("img/s.bmp");

t=SDL\_LoadBMP("img/t.bmp");

w=SDL\_LoadBMP("img/w.bmp");

u=SDL\_LoadBMP("img/u.bmp");

v=SDL\_LoadBMP("img/v.bmp");

x1=SDL\_LoadBMP("img/x.bmp");

y1=SDL\_LoadBMP("img/y.bmp");

z=SDL\_LoadBMP("img/z.bmp");

q=SDL\_LoadBMP("img/q.bmp");

ok=SDL\_LoadBMP("img/ok.bmp");

ca=SDL\_LoadBMP("img/ca.bmp");

go=SDL\_LoadBMP("img/go.bmp");

if (image == NULL)

{

printf("Unable to load bitmap: %s\n", SDL\_GetError());

return 1;

}

int x,y;

int done=1,aa=0,bb=0,cc=0,dd=0,ee=0,ff=0,gg=0,hh=0,i1=0,j1=0,kk=0,mm=0,nn=0,ll=0,oo=0,pp=0,qq=0,rr=0,ss=0,tt=0,uu=0,vv=0,ww=0,xx=0,yy=0,zz=0;

src.x = 0;

src.y = 0;

src.w = image->w;

src.h = image->h;

dest.x = 0;

dest.y = 0;

dest.w = image->w;

dest.h = image->h;

dest1.y=81;

dest2.x=347;

dest11.y=235;

dest22.x=335;

dest3.x=386;

dest3.y=153;

dest4.x=512;

dest4.y=343;

dest5.x=511;

dest5.y=156;

dest6.x=391;

dest6.y=342;

dest7.x=510;

dest7.y=154;

dest8.x=389;

dest8.y=339;

dest9.x=384;

dest9.y=151;

dest10.x=510;

dest10.y=341;

dest12.x=289;

dest12.y=374;

SDL\_BlitSurface(image, &src, screen, &dest);

while(done)

{

//SDL\_BlitSurface(image, &src, screen, &dest);

while(SDL\_PollEvent(&event))

{

switch (event.type)

{

case SDL\_MOUSEBUTTONUP:

x= event.button.x;

y= event.button.y;

break;

case SDL\_QUIT:

done = 0;

break;

case SDL\_KEYDOWN:

{

switch(event.key.keysym.sym)

{

case SDLK\_ESCAPE: /\* Appui sur la touche Echap, on arrête le programme \*/

done = 0;

break;

case SDLK\_a:

aa=1;

break;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*

case SDLK\_b:

bb=1;

break;

case SDLK\_c:

cc++;

break;

case SDLK\_d:

dd++;

break;

case SDLK\_e:

ee++;

break;

case SDLK\_f:

ff++;

break;

case SDLK\_g:

gg++;

break;

case SDLK\_h:

hh++;

break;

case SDLK\_i:

i1++;

break;

case SDLK\_j:

j1++;

break;

case SDLK\_k:

kk++;

break;

case SDLK\_l:

ll++;

break;

case SDLK\_m:

mm++;

break;

case SDLK\_n:

nn++;

break;

case SDLK\_o:

oo++;

break;

case SDLK\_p:

pp++;

break;

case SDLK\_q:

qq++;

break;

case SDLK\_r:

rr++;

break;

case SDLK\_s:

ss++;

break;

case SDLK\_t:

tt++;

break;

case SDLK\_u:

uu++;

break;

case SDLK\_v:

ww++;

break;

case SDLK\_w:

ww++;

break;

case SDLK\_x:

xx++;

break;

case SDLK\_y:

yy++;

break;

case SDLK\_z:

zz++;

break;

} //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

}

}

if((x>389)&&(x<408)&&(y>152)&&(y<174))

{

SDL\_SetColorKey(ok,SDL\_RLEACCEL | SDL\_SRCCOLORKEY,SDL\_MapRGB(ok->format,255,255,255));

SDL\_BlitSurface(ok, &src, screen, &dest3);

SDL\_BlitSurface(ok, &src, screen, &dest4);

SDL\_BlitSurface(ca, &src, screen, &dest7);

SDL\_BlitSurface(ca, &src, screen, &dest8);

}

else

if((x>514)&&(x<535)&&(y>156)&&(y<176))

{

SDL\_SetColorKey(ok,SDL\_RLEACCEL | SDL\_SRCCOLORKEY,SDL\_MapRGB(ok->format,255,255,255));

SDL\_BlitSurface(ok, &src, screen, &dest5);

SDL\_BlitSurface(ok, &src, screen, &dest6);

SDL\_BlitSurface(ca, &src, screen, &dest9);

SDL\_BlitSurface(ca, &src, screen, &dest10);

}

if((x>367)&&(x<623)&&(y>81)&&(y<120)&&(ii<6))

{

if(aa==1)

{

ii++;

dest1.x=dest2.x+(ii\*32);

SDL\_BlitSurface(a, &src, screen, &dest1);

aa=0;

}

}

if((x>365)&&(x<625)&&(y>81)&&(y<274)&&(jj<6))

{

if(aa==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(a, &src, screen, &dest11);

aa=0;

}

if(bb==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(b, &src, screen, &dest11);

bb=0;

}

if(cc==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(c, &src, screen, &dest11);

cc=0;

}

if(dd==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(d, &src, screen, &dest11);

dd=0;

}

if(ee==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(e, &src, screen, &dest11);

ee=0;

}

if(ff==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(f, &src, screen, &dest11);

ff=0;

}

if(gg==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(g, &src, screen, &dest11);

gg=0;

}

if(hh==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(h, &src, screen, &dest11);

hh=0;

}

if(i1==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(i, &src, screen, &dest11);

i1=0;

}

if(j1==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(j, &src, screen, &dest11);

j1=0;

}

if(kk==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(k, &src, screen, &dest11);

kk=0;

}

if(ll==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(l, &src, screen, &dest11);

ll=0;

}

if(mm==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(m, &src, screen, &dest11);

mm=0;

}

if(nn==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(n, &src, screen, &dest11);

nn=0;

}

if(oo==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(o, &src, screen, &dest11);

oo=0;

}

if(pp==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(p, &src, screen, &dest11);

pp=0;

}

if(qq==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(q, &src, screen, &dest11);

qq=0;

}

if(rr==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(r, &src, screen, &dest11);

rr=0;

}

if(ss==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(s, &src, screen, &dest11);

ss=0;

}

if(tt==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(t, &src, screen, &dest11);

tt=0;

}

if(uu==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(u, &src, screen, &dest11);

uu=0;

}

if(vv==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(v, &src, screen, &dest11);

vv=0;

}

if(ww==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(w, &src, screen, &dest11);

ww=0;

}

if(xx==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(x1, &src, screen, &dest11);

xx=0;

}

if(yy==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(y1, &src, screen, &dest11);

yy=0;

}

if(zz==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(z, &src, screen, &dest11);

zz=0;

}

if(mm==1)

{

jj++;

dest11.x=dest22.x+(jj\*32);

SDL\_BlitSurface(m, &src, screen, &dest11);

mm=0;

}

}

if((x>288)&&(x<401)&&(y>375)&&(y<475))

{

SDL\_BlitSurface(go, &src, screen, &dest12);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

}

SDL\_Flip(screen);

}

SDL\_FreeSurface(image);

return(0);

}