

```

enum TypeBouton
{
    action1 = 0,
    action2,
    action3,
    ...
    actionnn
} bouton_action = action1; // action par default

//
-----
// menu
// GLUT menu callback function. Gestion du menu. Selection d'une action
-----

static void
menu (int item)
{
    bouton_action = static_cast<TypeBouton> (item);
    glutPostRedisplay ();
}

-----
// display  comme d'habitude + appel des fonctions du menu.
-----

static void display (void)
{
    ...
    switch (bouton_action)
    {
        case action1:
            appelfonctionaction1 ();
            break;
        case action2:
            appelfonctionaction2 ();
            break;
        ...

        case actionnn:
            appelfonctionactionnn ();
            break;
    }
    ...
    glutSwapBuffers ();
}

// fonctions specifiques 1, 2, ...,n

static void appelaction1 (void)
{
    // action 1
}
static void appelaction1 (void)
{
    // action 2
}

```

```

...
static void appelaction1 (void)
{
    // action n
}

// Main comme d'hab + libelle menu

int main (int argc, char *argv[])
{
    // Initialisation glut
    ... comme d'habitude

    // Creation menu glut
    glutCreateMenu (menu);
    glutAddMenuEntry ("Libelle Action 1", action1);
    glutAddMenuEntry ("Libelle Action 2", action2);
    glutAddMenuEntry ("Libelle Action 3", action3);
    ...
    glutAddMenuEntry ("Libelle Action n", actionn);
    // par exemple si on veut faire apparaître le menu avec clic droit
    glutAttachMenu (GLUT_RIGHT_BUTTON);

    // glut callback functions
    ... comme d'habitude

    // main loop
    glutMainLoop ();
    return 0;
}

```