



Skill Test 5A : Dallage : TextBox



LAPINS NOIRS



Skill Test 5A Dallage

Sujet : TextBox

Le Laboratoire aux Lapins Noirs
lapinsnoirs@epitech.eu

Ce document contient un sujet d'examen associé au module d'infographie

Nom du dépôt de rendu : text_foot_promotion (Exemple flip_part_2042)



1 – Consignes

2 – Sujet



1 – Consignes

L'examen doit être réalisé avec la LibLapin comme unique outil. Les seules fonctions autorisées sont celles précisées dans la section sujet.

La compilation sera effectuée avec les flags -Wall -Wextra.

Si votre rendu comporte un fichier binaire, un fichier .o ou un fichier tampon type « `###` » ou « `*~` », vous obtiendrez la note de 1,5.

Si votre fonction ne s'appelle pas de la bonne façon, le programme de correction ne pourra pas la trouver et vous obtiendrez la note de 1,5.

Si votre programme est trop lent (>2 secondes), boucle à l'infinie, reçoit un signal SIGSEV, SIGFPE ou SIGPIPE, vous obtiendrez la note de 1,5.

La réussite de la compilation de votre programme avec la moulinette vous apporte la note de 2. Cette note évolue en fonction des résultats obtenus aux exercices.

Votre rendu ne doit pas comporter de main. Nous compilerons l'intégralité des fichiers .c rendus avec la moulinette. Les fichiers .h seront pris en compte si situé à la racine de votre rendu.



2 – Sujet

Preliminaire :

1 points

```
void                tekpixel(t_bunny_pixelarray    *pix,  
                           t_bunny_position        *pos,  
                           unsigned int           color) ;
```

Écrivez la fonction suivante, qui dessine un pixel de la couleur color à la position pos dans pix.



Skill Test 5A : Dallage : TextBox

Écrivez la fonction suivante :

2 points

```
void                                tekblit(t_bunny_pixelarray      *destination,  
                                             const t_bunny_pixelarray    *origin,  
                                             const t_bunny_position      *pos) ;
```

La fonction tekblit copie la zone définie par :

```
origin->clipable.clip_x_position  
origin->clipable.clip_y_position  
origin->clipable.clip_width  
origin->clipable.clip_height
```

à la position pos->x, pos->y, en prenant en compte les attributs origin.x et origin.y de l'image d'origine, dans l'image destination, le tout en redimensionnant correctement à l'aide des attributs scale.x et scale.y l'image.

Pour rappel :

Les attributs clip_* déterminent un sous ensemble du clipable à copier.

Les attributs scale contiennent une valeur à multiplier à la largeur et à la hauteur de la partie de l'image qu'on demande à copier.

les attributs origin.x et origin.y modifient le point d'ancrage de l'image, si par exemple, on demande à blitter un clip avec une origine a 50, 50 à la position 50, 50 dans une image, l'image sera affichée depuis la position 0, 0.

L'attribut rotation peut-être ignoré.

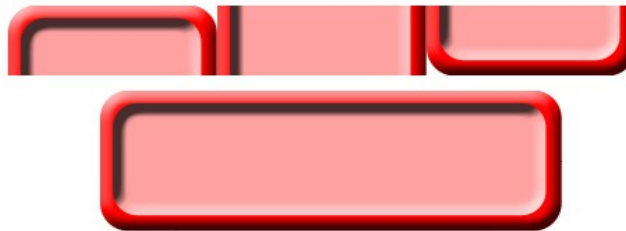


Écrivez la fonction suivante :

15 points

```
typedef struct                                s_bunny_textbox
{
    t_bunny_pixelarray    *chipset;
    int                    border_size;
    t_bunny_pixelarray    *font;
    t_bunny_position      fontsize;
    int                    x;
    int                    y;
    int                    w;
    int                    h;
}
t_bunny_textbox;

void                    textbox(t_bunny_pixelarray    *pix,
                                const t_bunny_textbox    *box,
                                const char                *str) ;
```



La fonction `textbox` affiche une boîte de texte stylisé à l'aide d'une planche de décor passé en paramètre via l'attribut `chipset` de `box`. La `textbox` doit être positionné à `x`, `y` et faire `w` et `h` de largeur et de hauteur.

Les attributs `chipset->clipable.clip_width` et `chipset->clipable.clip_height` vous indiqueront la taille d'une dalle (ou « tile ») dans le fichier. Le format graphique du fichier sera identique à celui présenté ci-dessus. D'abord le coin supérieur gauche, puis la bordure supérieure, etc.

L'attribut `border_size` de `box` vous indiquera quelle est la zone de la `textbox` dans laquelle vous pourrez écrire. Vous devez bien sûr commencer le plus en haut à gauche possible.

L'attribut `font` est un fichier de font chargé, contenant une table `ascii` et dont les caractères font une taille indiquée dans `fontsize`.

Les attributs `scale` de `chipset` comme de `font` ne doivent pas être ignoré. Vous devez les appliquer.



Skill Test 5A : Dallage : TextBox

Si une lettre est susceptible de dépasser sur la droite de la textbox, vous ne devez pas l'afficher mais sauter une ligne. Si des caractères sont susceptibles de dépasser vers le dessous de la textbox, vous devez l'afficher en partie (La partie supérieure, celle qui ne dépasse pas).

Aucun test avec une textbox inférieure à deux fois la largeur et hauteur d'une dalle ne seront fait.

...n'oubliez pas de déclarer la structure dans votre fichier...