**Báo Cáo Bài Tập Lớn**

Môn Học: Đồ Họa Máy Tính

Chủ Đề: Trò Chơi Toán Học

Thành Viên:

Nguyễn Văn Hợp

Vương Thị Hồng

Ngô Thị Oanh

Nguyễn Thu Trang

Sầm Đức Vũ

**I. Chủ Đề**

Trò Chơi Toán Học

**II. Ý Tưởng – Cách Chơi**

**- Ý tưởng:**

Mỗi lượt chơi, người chơi sẽ có 300s. Máy tính sẽ đưa ra ngẫu nhiên 4 số tự nhiên (nhỏ hơn 100), bằng các phép toán cộng, trừ, nhân, chia, người chơi cần tính toán để đưa ra kết quả cuối cùng là 10. Nếu sau 3 phép tính, kết quả người chơi tính được là 10 thì sẽ được cộng 1 điểm thưởng. Nếu kết quả cuối cùng đưa ra khác 10, người chơi không được cộng điểm.

Với mỗi 1 điểm thưởng, người chơi có thêm 1 cơ hội đổi 4 số tự nhiên khác cho các vòng tiếp theo. Lượt chơi sẽ kết thúc sau 300s.

**- Cách chơi:**

Người chơi dùng chuột để kéo thả các số tự nhiên vào ô trống, chọn phép tính, khi có đủ 2 số tự nhiên trong 2 ô trống, nếu đó là phép tính thực hiện được(phép cộng và phép nhân trong phạm vi 100, phép chia ra số nguyên và phép trừ ra một số không âm) máy tính sẽ hiện ra nút GO và khi người chơi chọn GO, một số mới sẽ được sinh ra(kết quả của phép tính).

Để chọn hiện thị bóng hay không, người chơi có thể nhấm phím A và S.

**III. Thực hiện**

**1. Kéo thả các ô vuông lưu số tự nhiên**

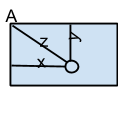
Mỗi số tự nhiên được lưu trên một button.

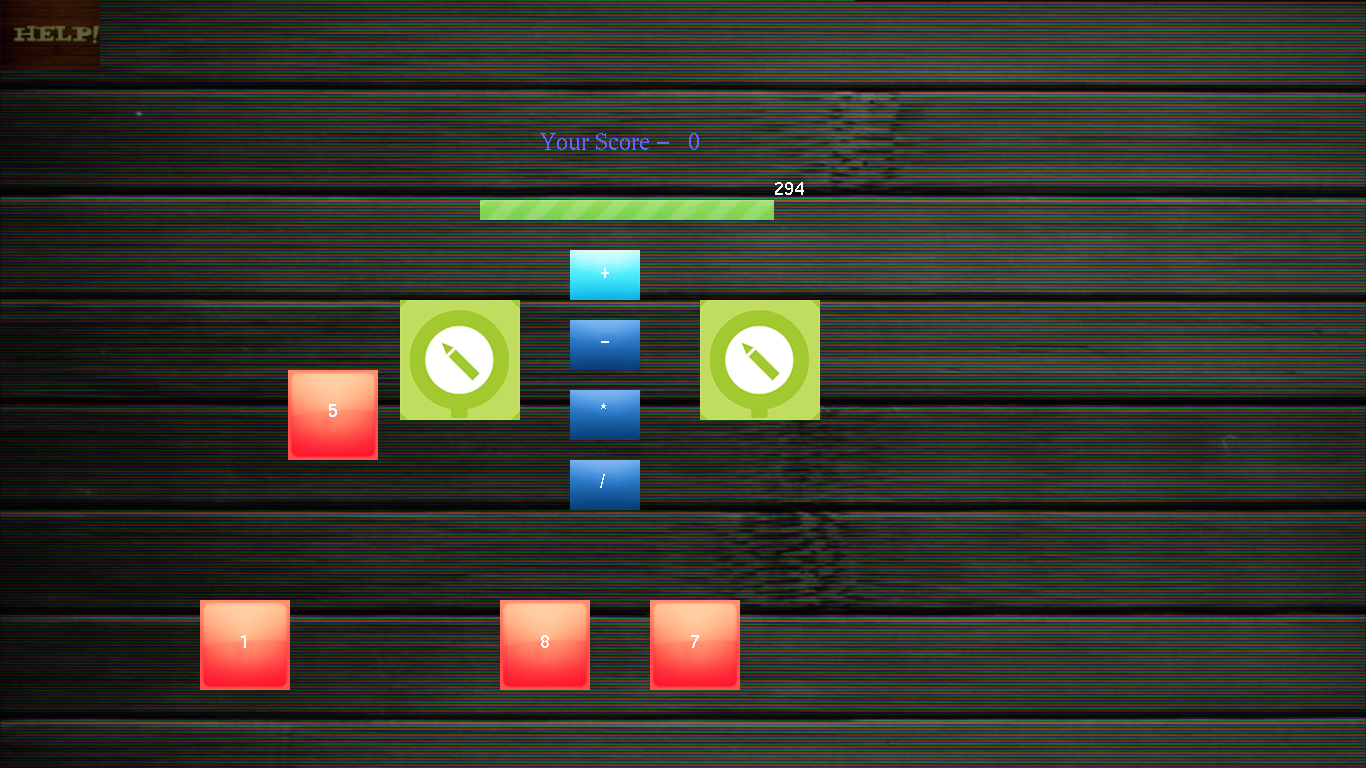
- Ta có một mảng cố định 4 phần tử lưu trữ tọa độ của 4 điểm, xác đinh 4 button của các số tự nhiên, mảng này là cố định và tọa độ các điểm xác định cũng là cố định

- Ta cần 1 vector lưu trữ các button chứa các giá trị các số. Vì số lượng các button luôn thay đổi sau khi ta kéo thả để thực hiện các phép toán, hoặc khi sang một level mới nên việc lưu trữ button bằng vector sẽ rất thuận tiện cho việc thêm, bớt các button.

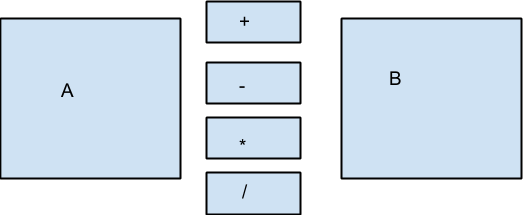
- Thực tế của việc kéo thả các button là di chuyển chúng ra một vị trí khác và vẽ lại chúng.

Khi click chuột vào một button ta sẽ đo khoảng các x, y(như hình vẽ). Từ x, y ta tính được z. Để có thể kéo được button ra vị trí khác ta phải xóa button tại vị trí cũ đi và thực hiện vẽ lại một button tại ví trí mới. Dựa vào z. ta sẽ xác định được đinh của button(A), và từ đó vẽ lại button dựa trên đỉnh A này.





- Khi thả button vào ô để thực hiện tính toán.

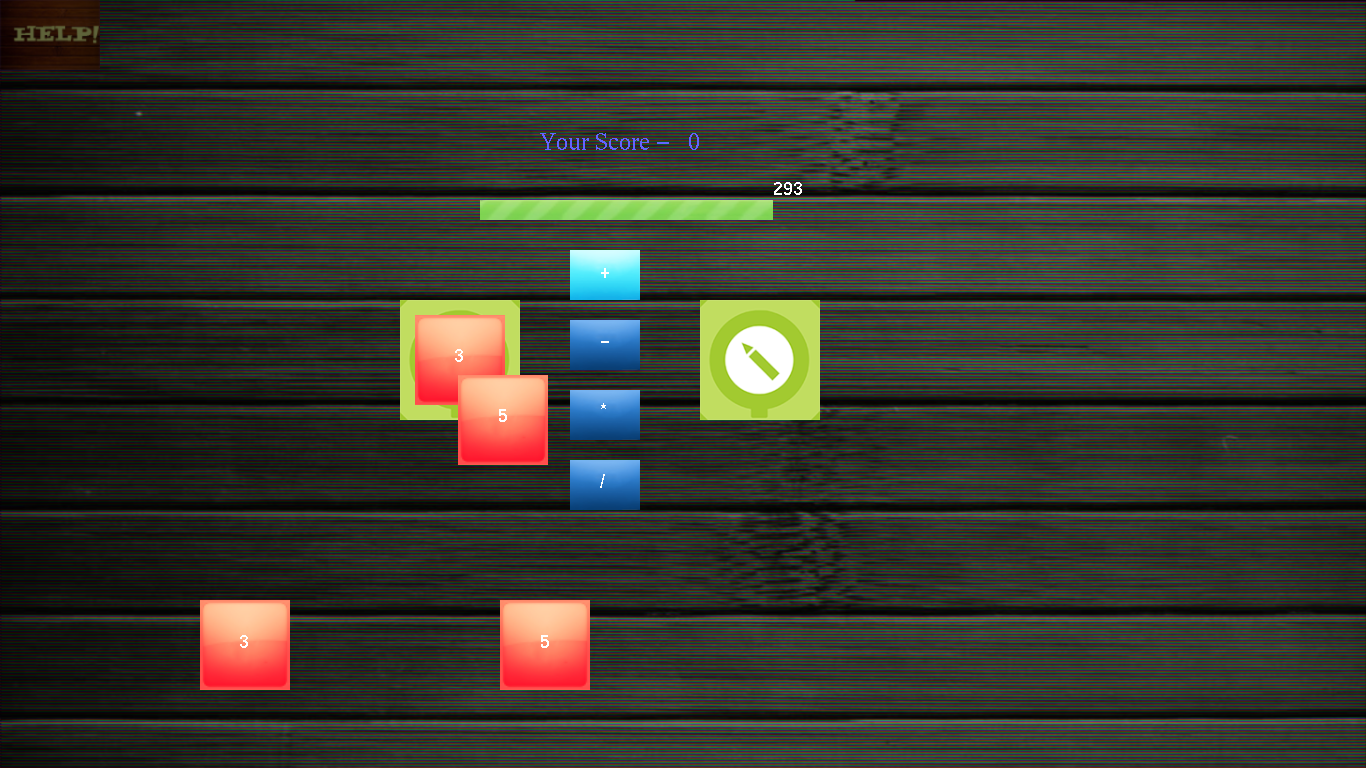


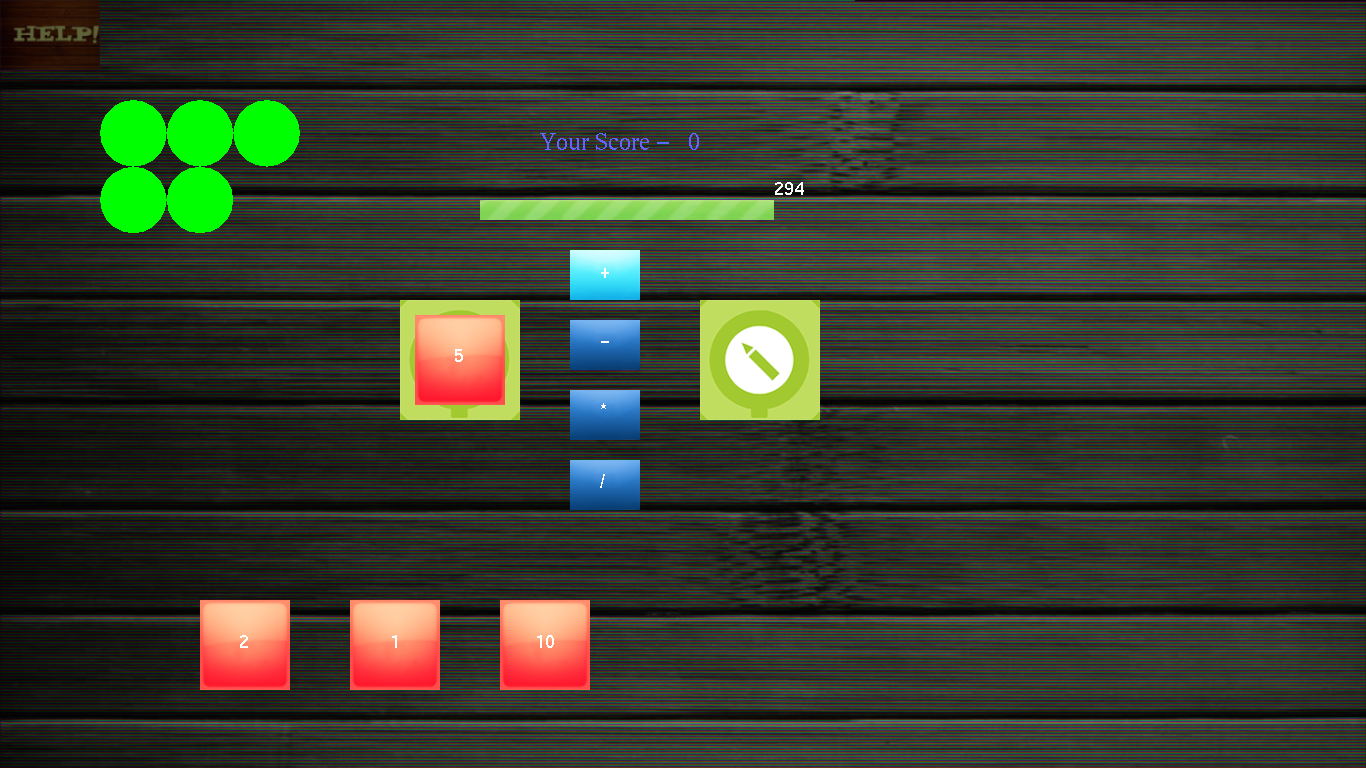
* Nếu trong ô A(hoặc B)chưa có môt button nào khác:

- Khi đặt chuột vào 1 vị trí trong button và thực hiện kéo thả, lúc này vector chứa các button sẽ được giảm đi một phần tử đồng thời cập nhật lại vector, nhưng mảng chứa tọa độ button thì không thay đổi.

-Khi button này được xác định nằm trong khung của A(hoặc B). Trong A(hoặc B) luôn xác định 1 button rỗng và khi đó Button rỗng này được gán bằng Button chứa số tự nhiên ta vừa kéo thả.

* Nếu trong ô A (hoặc B) đã có một button khác:
* Button đã có trong ô A(hoặc B)được thêm vào vị trí cuối của vector lưu các button.
* Thêm button mới vào ô A hoặc (B)như trường hợp trên





* Khi thả Button ra ngoài không (Không thả vào trong ô A hoặc B)

Button sẽ được trả về vị trí cũ của chính nó

* Khi phép toán được thực hiện, kết quả của phép toán sẽ được lưu trong một button mới và được thêm vào cuối vector chứa các button.

1. **Thuật toán hiện bóng**

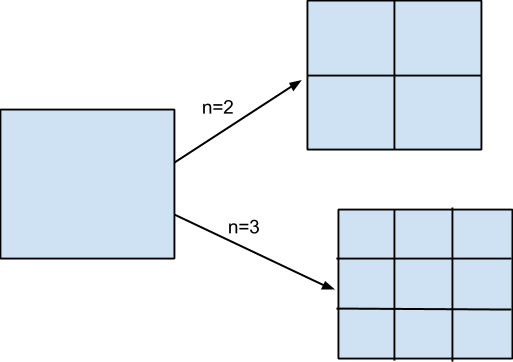
Để trò chơi thêm hấp dẫn, mỗi button khi được thả vào ô A hoặc B sẽ có một số lượng bóng tương ứng với giá trị của button hiện ra.

Kích thước của số lượng quả bóng là như nhau.

Xuất phát từ ý tưởng chia cửa sổ thành những ô có kích thước đều nhau

Giả sử xét cửa sổ hình vuông ban đầu có cạnh a , chia cửa sổ hình vuông này thành nhiều cửa sổ nhỏ hình vuông theo quy luật ( n [1;10]). Mỗi hình vuông nhỏ được chia sẽ chứa 1 hình tròn với bán kính =

Ví dụ :



Để chia được bao nhiêu ô hình vuông nhỏ thì dựa vào số nhập tính toán b. Theo quy luật như sau : (1)

Với : b =1 => n=1.

1< b ≤ 4 => n=2

4< b ≤ 9 => n=3

9< b ≤ 16 => n=4

16< b ≤ 25 => n=5

25< b ≤ 36 => n=6

36< b ≤ 49 => n=7

49< b ≤ 64 => n=8

64< b ≤ 81 => n=9

81< b ≤ 100 => n=10

Sau đó mỗi ô nhỏ sẽ chứa 1 hình tròn có đường kính là cạnh hình vuông nhỏ.

**Thuật toán :**

- Với số nhập tính toán b, dựa vào (1) ở trên ta chia được thành hình vuông nhỏ có kích thước cạnh là .

+Để vẽ đủ hình tròn với số lượng là b, đầu tiên ta vẽ 1 hình tròn tại 1 hình vuông được đánh dấu là điểm bắt đầu.

+ Sau đó, tịnh tiến tọa độ tâm hình tròn sang tọa độ chính giữa của hình vuông tiếp theo bên phải, ở đây vẽ tiếp hình tròn ở ô này,,,,, khi tịnh tiến tới số hình tròn = n : (2)

+ Trường hợp 1: Số bóng =n =b thì dừng lại.

+ Trường hợp 2 : số bóng được vẽ mà vẫn nhỏ hơn b thì quay lại tọa độ tâm hình tròn ban đầu, rồi tiếp tục tịnh tiến xuống tọa độ chính giữa của hình vuông dưới hình vuông ban đầu, rồi tiếp tục lặp lại bước (2) khi tổng số bóng = b thì dừng lại. (3)

1. **Thuật toán random**

Với mỗi lượt chơi có thể có nhiều vòng, và để 4 sô tự nhiên ngẫu nhiên cho mỗi vòng chương trình đã sinh ra các số tự nhiên ngẫu nhiên có 2 chữ số dựa và thật toán sau.

* Xuất phát từ kết quả cần tính là 10,
* (1) Chọn ngẫu nhiên một phép toán
* (2)sinh ra 2 số tự nhiên ngẫu nhiên dưa vào phép toán (1)

ví dụ: phép toán là + , 2 số ngẫu nhiên sinh ra là 0 và 10 hoặc 5 và 5 hoặc 4 và 6…

* (3)Chọn ngẫu nhiên 1 trong 2 số vừa sinh ra trong (2)
* (4) Chọn một phép tính ngẫu nhiên, sinh ra 2 số ngẫu nhiên từ (3) và phép toán này
* (5)Chọn 1 số ngẫu nhiên trong 3 số đã sinh ra
* (6)Chọn 1 phép toán ngẫu nhiên, sinh ra 2 số ngẫu nhiên từ (5) và phép toán này

Ta đã sinh ra được 4 số ngẫu nhiên dựa trên kết quả mong muốn cuối cùng là 10.

1. **Dán ảnh chất liệu**

Để trò chơi thêm sinh động, hấp dẫn, nhóm em đã dán ảnh chất liệu lên các button , hình nền…

Nhóm đã sử dụng ảnh bitmap để dán với button có cỡ 256x256, và background có cỡ 1366x768

Riêng đối với các button hiển thị phép tính(cộng, trừ, nhân, chia) cho phép dán ảnh chất liệu khác khi được click chọn để hiển thị phép tính do người chơi chọn khác với chất liệu ảnh mặc định.

Các bước thực hiện để dán ảnh chất liệu cho một đối tượng thì ta sẽ thực hiện qua các bước cơ bản sau:

* Load ảnh bitmap từ một file .bmp

Khi load một ảnh bitmap ta cần xác định rõ các thông tin chiều rộng, chiều cao, các bit trên mỗi điểm ảnh của ảnh.

* Sau khi mở file ta tiến hành đọc các thông tin về chiều cao, chiều rộng của ảnh vào cấu trúc BITMAPINFOHEADER.
* Xác định số bit thực tế trên ảnh
* Mở file theo chế độ nhị phân và trả về NULL nếu không có file được mở

if( (file = fopen(filename, "rb"))==NULL) return (-1);

- Di chuyển con trỏ đến bit 18 và đọc các thông tin về chiều rộng, chiều cao, số mặt phẳng của ảnh

fseek(file, 18, SEEK\_CUR);

fread(&infoheader.biWidth, sizeof(int), 1, file);

fread(&infoheader.biHeight, sizeof(int), 1, file);

fread(&infoheader.biPlanes, sizeof(short int), 1, file);

- Đọc số bit của ảnh

fread(&infoheader.biBitCount, sizeof(unsigned short int), 1, file);

* Cấp phát bộ nhớ cho dữ liệu ảnh

infoheader\_data = (unsigned char \*) malloc(infoheader.biWidth \* infoheader.biHeight \* 3);

* Các ảnh bitmap lưu trữ các điểm ảnh theo định dạng BGR, ta phải chuyển nó về định dạng RGB

for (i=0; i<(infoheader.biWidth \* infoheader.biHeight \* 3); i+=3) { // reverse all of the colors. (bgr -> rgb)

temp = infoheader\_data[i];

infoheader\_data[i] = infoheader\_data[i+2];

infoheader\_data[i+2] = temp;

}

* Đặt tên cho chất liệu đồng thời xác định số chất liệu mà mình muốn tạo ra bằng hàm

glGenTextures(1, &num\_texture);

* Lựa chọn chất liệu muốn thiết lập bằng hàm

glBindTexture(GL\_TEXTURE\_2D, num\_texture);

* Thiết lập các thuộc tính cho chất liệu đã chọn và xác định chất liệu là các kết cấu 2D

glTexParameterf(GL\_TEXTURE\_2D, GL\_TEXTURE\_WRAP\_S, GL\_REPEAT);

glTexParameterf(GL\_TEXTURE\_2D, GL\_TEXTURE\_WRAP\_T, GL\_REPEAT);

glTexParameterf(GL\_TEXTURE\_2D,

GL\_TEXTURE\_MAG\_FILTER,GL\_LINEAR);

glTexParameterf(GL\_TEXTURE\_2D, GL\_TEXTURE\_MIN\_FILTER, GL\_LINEAR\_MIPMAP\_NEAREST);

glTexEnvf(GL\_TEXTURE\_ENV, GL\_TEXTURE\_ENV\_MODE, GL\_MODULATE);

* Định nghĩa dán ảnh chất liệu 2D bằng hàm

glTexImage2D(GL\_TEXTURE\_2D, 0, 3, infoheader.biWidth, infoheader.biHeight, 0, GL\_RGB, GL\_UNSIGNED\_BYTE, infoheader\_data);

* Hàm Load sẽ được gọi trong các class để dán ảnh chất liệu cho các Block, Button, …..

1. Phân Công Công Việc
2. Texture + hàm random 4 số tự nhiên ngẫu nhiên: Oanh, Hồng, Trang
3. Hiện Bóng: Vũ
4. Kéo thả button: Hợp
5. Slide: Hợp
6. Báo Cáo: Hồng, Oanh, Trang