

Tạo kẻ thù

Thực hiện:

Phát triển tiếp từ bài 4:

Ta sẽ tạo ra các kẻ thù được chạy ra từ bên phải-> trái, các kẻ thù này có hình khác nhau và tốc độ chạy cũng khác nhau. (thật ra nó hơi giống với lớp viên đạn).

Phần 1: Tạo class Enemies (viên đạn).

1. Tạo một class mới tên Enemies.
2. Trong class vừa tạo tạo ra các biến toàn cục như mã dưới. Biến gồm x,y là tọa độ của kẻ thù, hình của kẻ thù và đặc biệt là 2 mảng tốc độ và mảng hình vì mỗi lần tạo một đối tượng kẻ thù ta sẽ lấy ngẫu nhiên để ra một hình và tốc độ của nó trong mảng. Biến e_ngnhien chính là tạo ngẫu nhiên 1 số rồi lấy hình và tốc độ để hình thành nên kẻ thù.

```
int x;
int y;
Bitmap bitmap;
int []tocdo={10,20,50};
int []manghinh={R.drawable.cam,
                R.drawable.le,
                R.drawable.tao};
int e_ngnhien;
```

3. Xây một hàm tạo nhận vào resource và 2 biến chứa chiều rộng và chiều cao của canvas (để ta lấy trục x). Trong hàm ta tạo ngẫu nhiên sau đó lấy hình theo ngẫu nhiên. Tọa độ x của kẻ thù bằng chiều rộng canvas (bắt đầu từ phải), tọa độ y là lấy ngẫu nhiên trong chiều cao canvas. Trong hàm doDraw ta sẽ vẽ đối tượng và mỗi lần vẽ ta sẽ cập nhật lại trục x, bằng cách giảm x đi (vì chạy từ phải -> trái). Các hàm còn lại linh tinh không dùng đến.

```
public Enemies(Resources res,int rong_cv,int cao_cv)
{
    Random rand=new Random();
    e_ngnhien=rand.nextInt(3);
    Log.d("nn", ""+e_ngnhien);
    this.x=rong_cv;//x tu phai
    int a=0+(int) (Math.random()*((cao_cv-0)+1));
    this.y=a;
    bitmap=BitmapFactory.decodeResource(res,manghinh[e_ngnhien]);
}
public Enemies(Resources res,int x,int y, int hinh)
{
    this.x=x;
    this.y=y;
    bitmap=BitmapFactory.decodeResource(res, hinh);
}
public void doDraw(Canvas canvas)
{
    canvas.drawBitmap(bitmap, x,y, null);
    x-=tocdo[e_ngnhien]; //tru vi chay phai sang trai
}
public void setXY(int x,int y)
{
    this.x=x;
```

```
        this.y=y;
    }
```

Phần 2 : viết trên GamePanel.java

1. Quay về file “GamePanel.java” và khai báo các biến toàn cục.

```
ArrayList<Enemies> enemies=new ArrayList<Enemies>();
int thoigiankethu=0;//thoi gian ra ke thu, 10 se ra
Enemies motkethu;
```

2. Trong hàm doDraw ta xây một hàm mới tên doDrawEnemies. Hàm này sẽ tạo ra kẻ thù và thêm nó vào ArrayList tên enemies. Sau đó dùng for để vẽ lại từng kẻ thù một, sau đó cũng dùng for để duyệt tất cả kẻ thù, kẻ thù nào đã ra khỏi canvas thì xóa nó khỏi ArrayList.

```
//ve tap hop ke thu
public void doDrawEnemies(Canvas canvas)
{
    if(thoigiankethu>=10)
    {
        thoigiankethu=0;
        Enemies motkethu=new Enemies(getResources(),
            canvas.getWidth(),canvas.getHeight());
        enemies.add(motkethu);
    }
    for(int i=0;i<enemies.size();i++)
        enemies.get(i).doDraw(canvas);

    for(int i=0;i<enemies.size();i++)
        if(enemies.get(i).y<0)
            enemies.remove(i);
    Log.d("viendan", "so vien: "+enemies.size());
}
```

3. Quay lên hàm onDraw ta bổ sung thêm lệnh để tăng thời gian kẻ thù lên, và gọi hàm doDrawEnemies để vẽ kẻ thù ra, trong hàm vẽ sẽ xét biến thời gian kẻ thù để vẽ kẻ thù ra.

```
.....
        thoigiannapdan++;
        thoigiankethu++;

        if(myelement!=null)
        {
            myelement.doDraw(canvas);//ve may bay
            this.doDrawBullet(canvas); //ve tap hop vien dan
            this.doDrawEnemies(canvas);//ve tap hop Enemies
        }
    }
    .....
}
```