

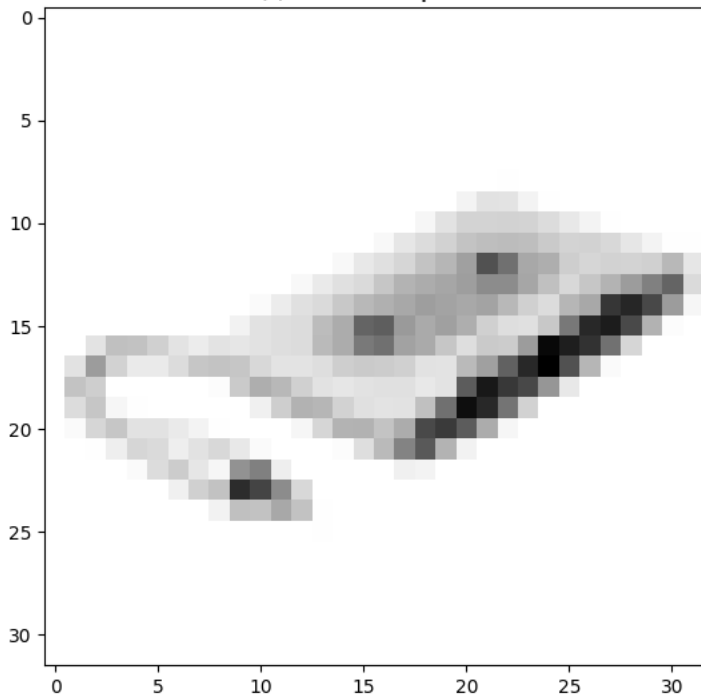
9. Определите на первоначальной тестовой выборке изображение, имеющее наибольшую ошибку реконструкции. Выведите для этого изображения первоначальное и реконструированное изображения.

```
[94] X = X2.reshape(X2.shape[0], 32, 32)
X = X.astype('float32') / 255.
encoded_imgs = encoder.predict(X2)
decoded_imgs = decoder.predict(encoded_imgs)
```

```
25/25 [=====] - 0s 4ms/step
25/25 [=====] - 0s 4ms/step
```

```
RE = ((X2 - decoded_imgs)**2).mean(axis = 1)
RE_original = RE.copy()
biggest_re_pos = np.argmax(RE_original)
fig = plt.figure(figsize = (14, 7))
ax = fig.add_subplot(1, 2, 1)
plt.title('Исходное изображение', fontsize = 16)
plt.imshow(X2[biggest_re_pos].reshape(32, 32), cmap='gray');
ax = fig.add_subplot(1, 2, 2)
plt.title('Реконструированное изображение', fontsize = 16)
plt.imshow(decoded_imgs[biggest_re_pos].reshape(32, 32), cmap='gray');
```

Исходное изображение



Реконструированное изображение

