* Clipboard là một vùng dữ liệu nhớ tạm trên RAM.
* Lấy văn bản nhập từ bàn phím là một nhiệm vụ cơ bản trong game. Ở đây, chúng ta sẽ lấy văn bản sử dụng cách nhập văn bản và xử lý clipboard mới của SDL 2.

|  |
| --- |
| // Main loop flag              bool quit = false;              // Event handler              SDL\_Event e;              // Set text color as black              SDL\_Color textColor = {0, 0, 0, 0xFF};              // The current input text.              std::string inputText = "Some Text";              gInputTextTexture.loadFromRenderedText(inputText.c\_str(), textColor);              // Enable text input              SDL\_StartTextInput(); |

* Trước khi chúng ta đi vào main loop, chúng ta khai báo một xâu để giữ văn bản của chúng ta và render nó lên một texture. Sau đó chúng ta gọi SDL\_StartTextInput để chức năng nhập văn bản của SDL được bật.

|  |
| --- |
| // While application is running              while (!quit)              {                  // The rerender text flag                  bool renderText = false;                  // Handle events on queue                  while (SDL\_PollEvent(&e) != 0)                  { |

* Chúng ta chỉ muốn update text texture khi chúng ta cần, vì thế chúng ta có một flag mà theo dõi xem chúng ta có cần update texture không.

|  |
| --- |
| // Special key input                      else if (e.type == SDL\_KEYDOWN)                      {                          // Handle backspace                          if (e.key.keysym.sym == SDLK\_BACKSPACE && inputText.length() > 0)                          {                              // lop off character                              inputText.pop\_back();                              renderText = true;                          }                          // Handle copy                          else if (e.key.keysym.sym == SDLK\_c && SDL\_GetModState() & KMOD\_CTRL)                          {                              SDL\_SetClipboardText(inputText.c\_str());                          }                          // Handle paste                          else if (e.key.keysym.sym == SDLK\_v && SDL\_GetModState() & KMOD\_CTRL)                          {                              inputText = SDL\_GetClipboardText();                              renderText = true;                          } |

* Có một vài cặp phím đặc biệt mà chúng ta muốn xử lý. Khi ng dùng nhấn Backspace, chúng ta cần xóa ký tự cuối cùng của xâu.
* Khi ng dùng dùng Ctrl + C, ta cần copy văn bản hiện tại vào clipboard sử dụng SDL\_SetClipboardText. Bạn có thể kiểm tra xem Ctrl có được giữ không sử dụng SDL\_GetModState.
* Khi chúng ta dùng Ctrl + V, chúng ta muốn lấy văn bản trong clipboard sử dụng SDL\_GetClipboardText. Cũng chú ý rằng, bất cứ khi nào chúng ta thay thế nội dung của xâu, chúng ta sẽ thiết lập flag cập nhật nhật văn bản.

|  |
| --- |
| // Special text input event                      else if (e.type == SDL\_TEXTINPUT)                      {                          // Not copy or pasting                          if (!(SDL\_GetModState() & KMOD\_CTRL && (e.text.text[0] == 'c' || e.text.text[0] == 'C' || e.text.text[0] == 'v' || e.text.text[0] == 'V')))                          {                              // Append character                              inputText += e.text.text;                              renderText = true;                          }                      }                  } |

* Khi nhập văn bản được bật, phím bạn nhấn sẽ sinh ra SDL\_TextInputEvent, cái sẽ giúp đơn giảm hóa những phím như SHIFT và CAPS LOCK. Ở đây chúng ta đầu tiên sẽ check xem có nhận được sự kiện Ctrl và C/V không vì chúng ta sẽ bỏ qua chúng vì đã xử lý rồi. Nếu không phải sụ kiện copy paste, chúng ta chèn thêm kí tự vào cuối input string của chúng ta.

|  |
| --- |
| // Rerender text if needed  if (renderText)  {      // Text is not empty      if (inputText != "")      {          // Render new text          gInputTextTexture.loadFromRenderedText(inputText.c\_str(), textColor);      }      // Text is empty      else      {          // Render space texture          gInputTextTexture.loadFromRenderedText(" ", textColor);      }  } |

* Nếu flag cập nhật render văn bản được cập nhật, chúng ta render cái texture. Một mẹo nhỏ chúng ta ở đây là nếu chúng ta có một xâu rỗng, chúng ta render một dấu cách vì SDL\_ttf sẽ không render một string rỗng.

|  |
| --- |
| // Clear screen                  SDL\_SetRenderDrawColor(gRenderer, 0xFF, 0xFF, 0xFF, 0xFF);                  SDL\_RenderClear(gRenderer);                  // Render text textures                  gPromptTextTexture.render((SCREEN\_WIDTH - gPromptTextTexture.getWidth()) / 2, 0);                  gInputTextTexture.render((SCREEN\_WIDTH - gInputTextTexture.getWidth()) / 2, gPromptTextTexture.getHeight());                  // Update screen                  SDL\_RenderPresent(gRenderer); |

* Ở cuối main loop, chúng ta render một câu hỏi và văn bản nhập vào.

|  |
| --- |
| // Disable text input              SDL\_StopTextInput(); |

Một khi xong phần nhập văn bản thì ta cần tắt nó.