Southern Luzon State University
College of Engineering
Computer Engineering Department

CPE13 Object Oriented Programming

Activity 9: Menu and Submenu

Name: REIMARC G. CORPUZ

Date: NOV 29, 2022

Section: BSCPE 3GF

Score:_______________

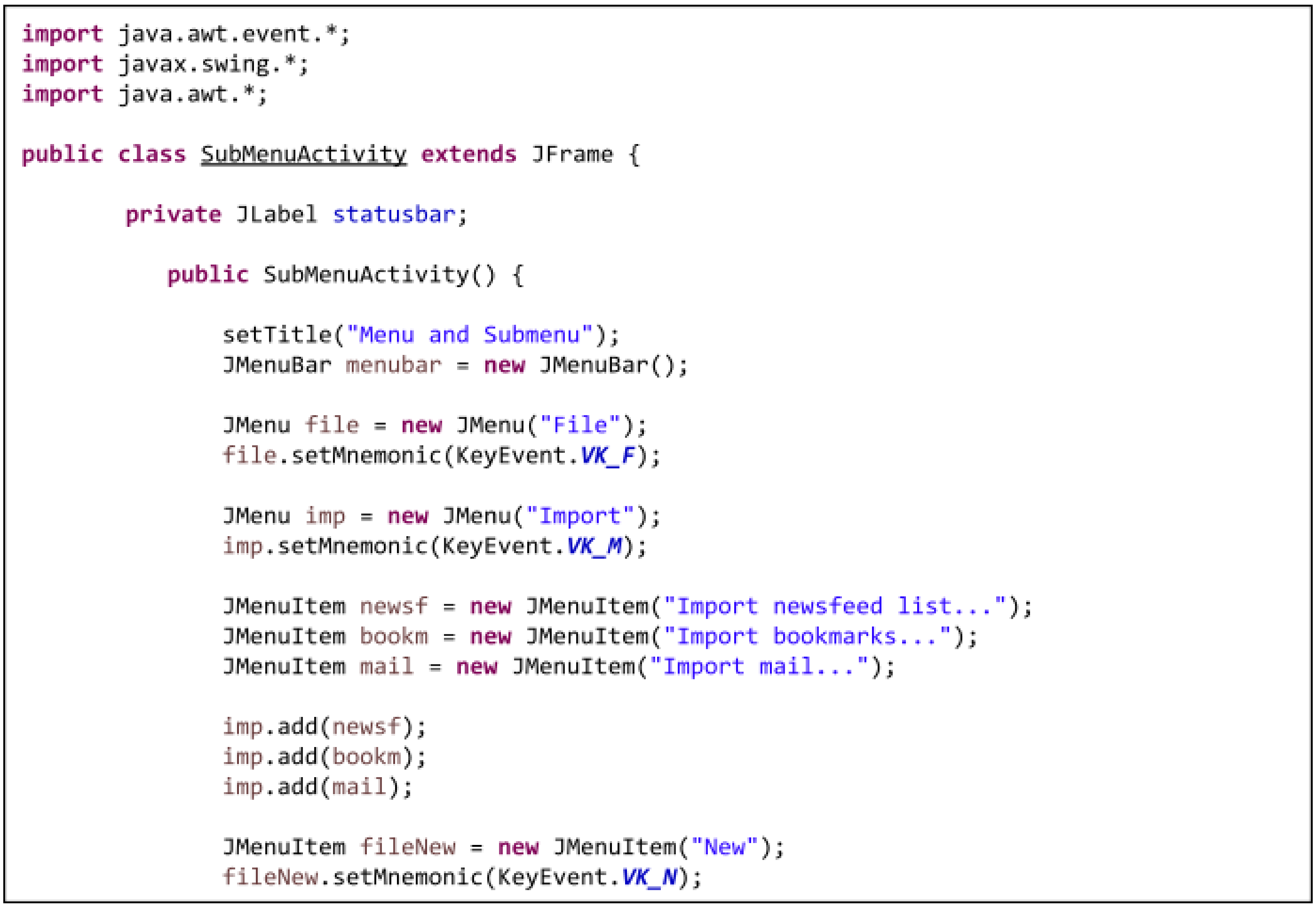
1.1 Introduction

A menubar is one of the most visible parts of the GUI application. It is a group of commands located in various menus. While in console applications you had to remember all those arcane commands, here we have most of the commands grouped into logical parts. There are accepted standards that further reduce the amount of time spending to learn a new application. In Java Swing, to implement a menubar, we use three objects. A JMenuBar, a JMenu and a JMenuItem.

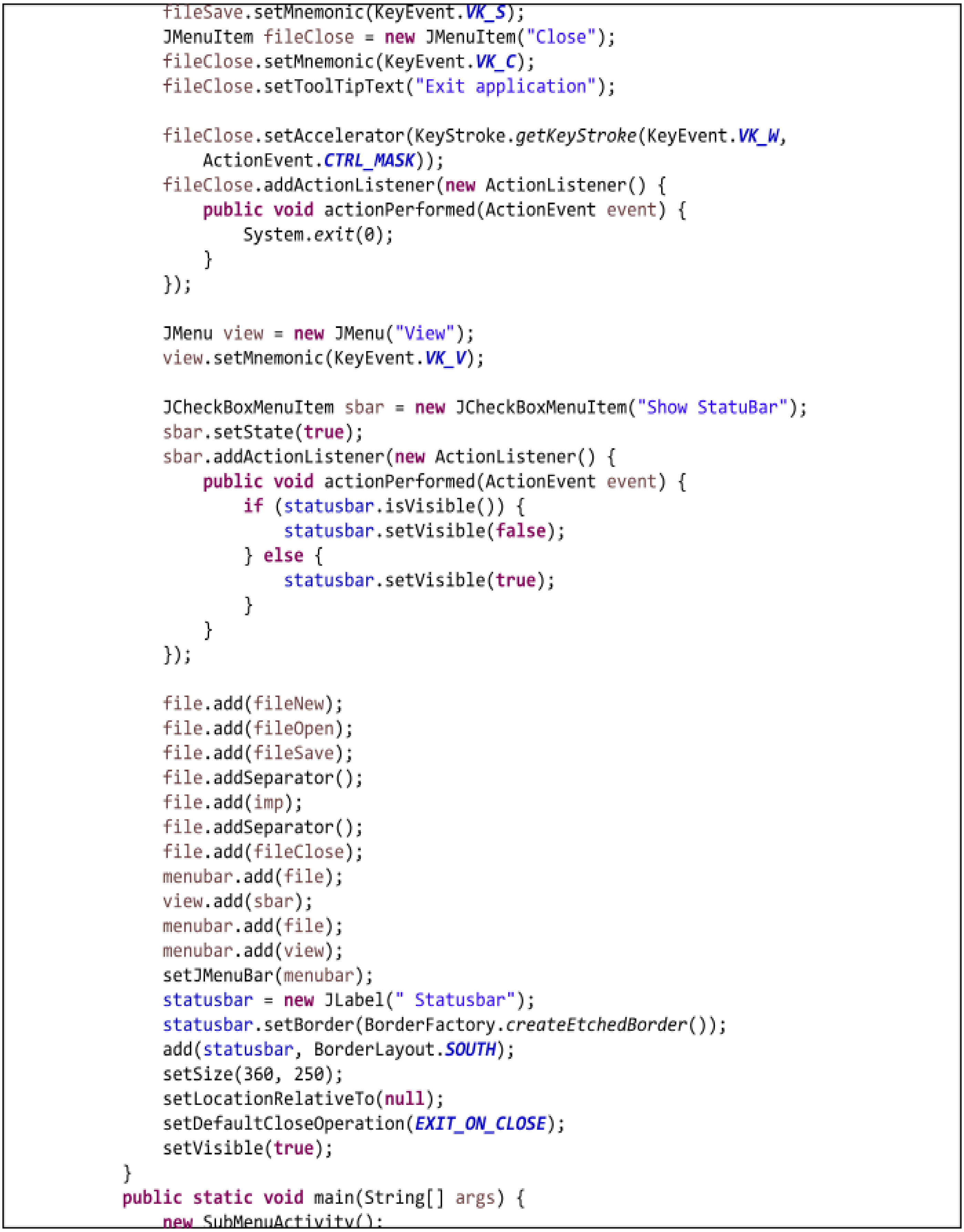
1.2 Objective

- To use Java programming language to create a program that uses Menu and sub menu.
- To conceptualize the process and manipulate the program
- To distinguish different parts of GUI Creation particularly the creation of menu and its properties.

Sample Program:



```
import java.awt.event.*;
import javax.swing.*;
import java.awt.*;

public class SubMenuActivity extends JFrame {

        private JLabel statusbar;

           public SubMenuActivity() {

               setTitle("Menu and Submenu");
               JMenuBar menubar = new JMenuBar();

               JMenu file = new JMenu("File");
               file.setMnemonic(KeyEvent.VK_F);

               JMenu imp = new JMenu("Import");
               imp.setMnemonic(KeyEvent.VK_M);

               JMenuItem newsf = new JMenuItem("Import newsfeed list...");
               JMenuItem bookm = new JMenuItem("Import bookmarks...");
               JMenuItem mail = new JMenuItem("Import mail...");

               imp.add(newsf);
               imp.add(bookm);
               imp.add(mail);

               JMenuItem fileNew = new JMenuItem("New");
               fileNew.setMnemonic(KeyEvent.VK_N);
```

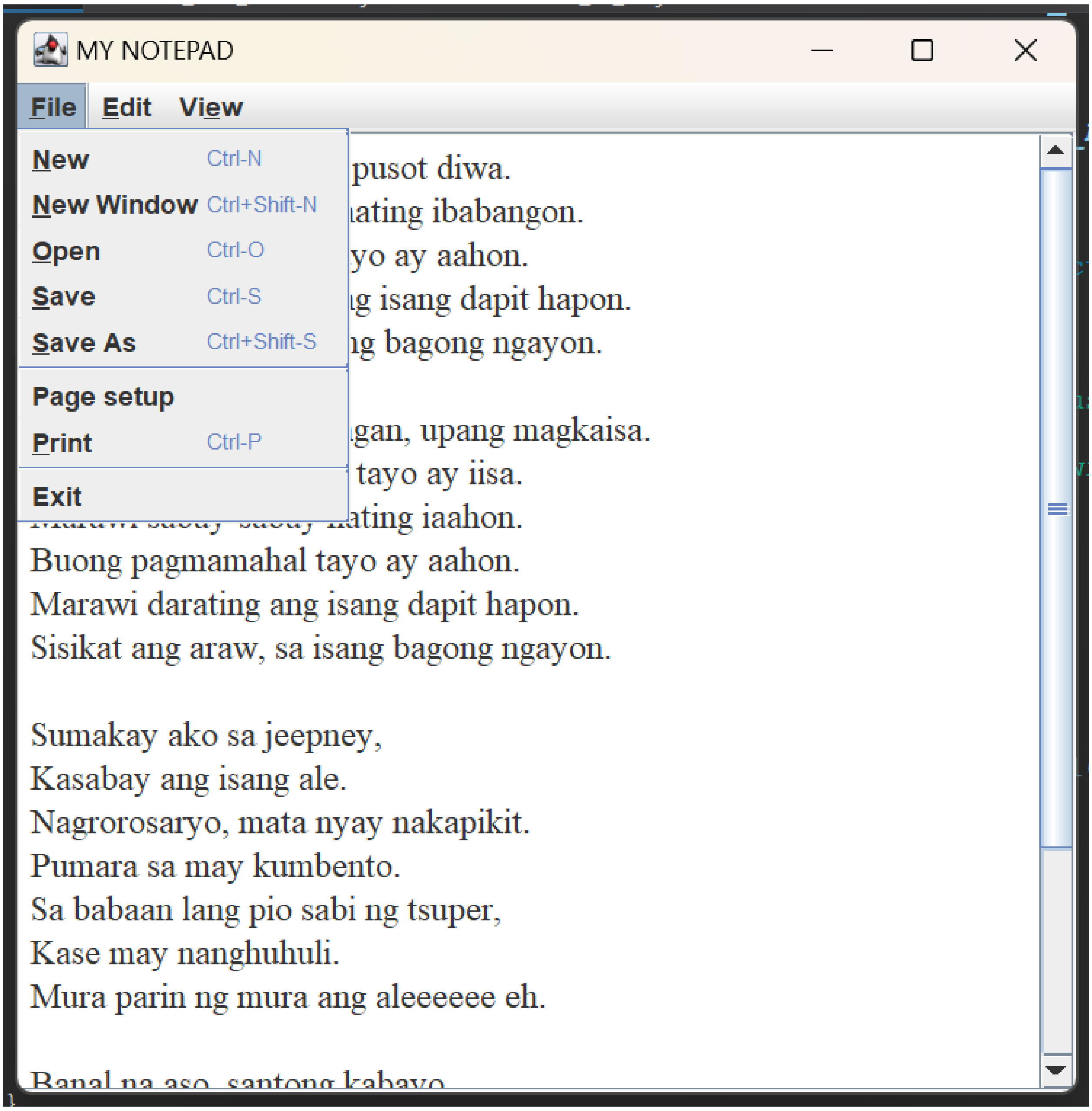


```
        fileSave.setMnemonic(KeyEvent.VK_S);
        JMenuItem fileClose = new JMenuItem("Close");
        fileClose.setMnemonic(KeyEvent.VK_C);
        fileClose.setToolTipText("Exit application");

        fileClose.setAccelerator(KeyStroke.getKeyStroke(KeyEvent.VK_W,
            ActionEvent.CTRL_MASK));
        fileClose.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent event) {
                System.exit(0);
            }
        });

        JMenu view = new JMenu("View");
        view.setMnemonic(KeyEvent.VK_V);

        JCheckBoxMenuItem sbar = new JCheckBoxMenuItem("Show StatuBar");
        sbar.setState(true);
        sbar.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent event) {
                if (statusbar.isVisible()) {
                    statusbar.setVisible(false);
                } else {
                    statusbar.setVisible(true);
                }
            }
        });

        file.add(fileNew);
        file.add(fileOpen);
        file.add(fileSave);
        file.addSeparator();
        file.add(imp);
        file.addSeparator();
        file.add(fileClose);
        menubar.add(file);
        view.add(sbar);
        menubar.add(file);
        menubar.add(view);
        setJMenuBar(menubar);
        statusbar = new JLabel(" Statusbar");
        statusbar.setBorder(BorderFactory.createEtchedBorder());
        add(statusbar, BorderLayout.SOUTH);
        setSize(360, 250);
        setLocationRelativeTo(null);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setVisible(true);
    }
    public static void main(String[] args) {
        new SubMenuActivity();
```

1.3 Problem

Write a program of Layout of a Notepad, all of its menu, sub menu and a text area with scroll.

Screenshot of the disp[lay output:

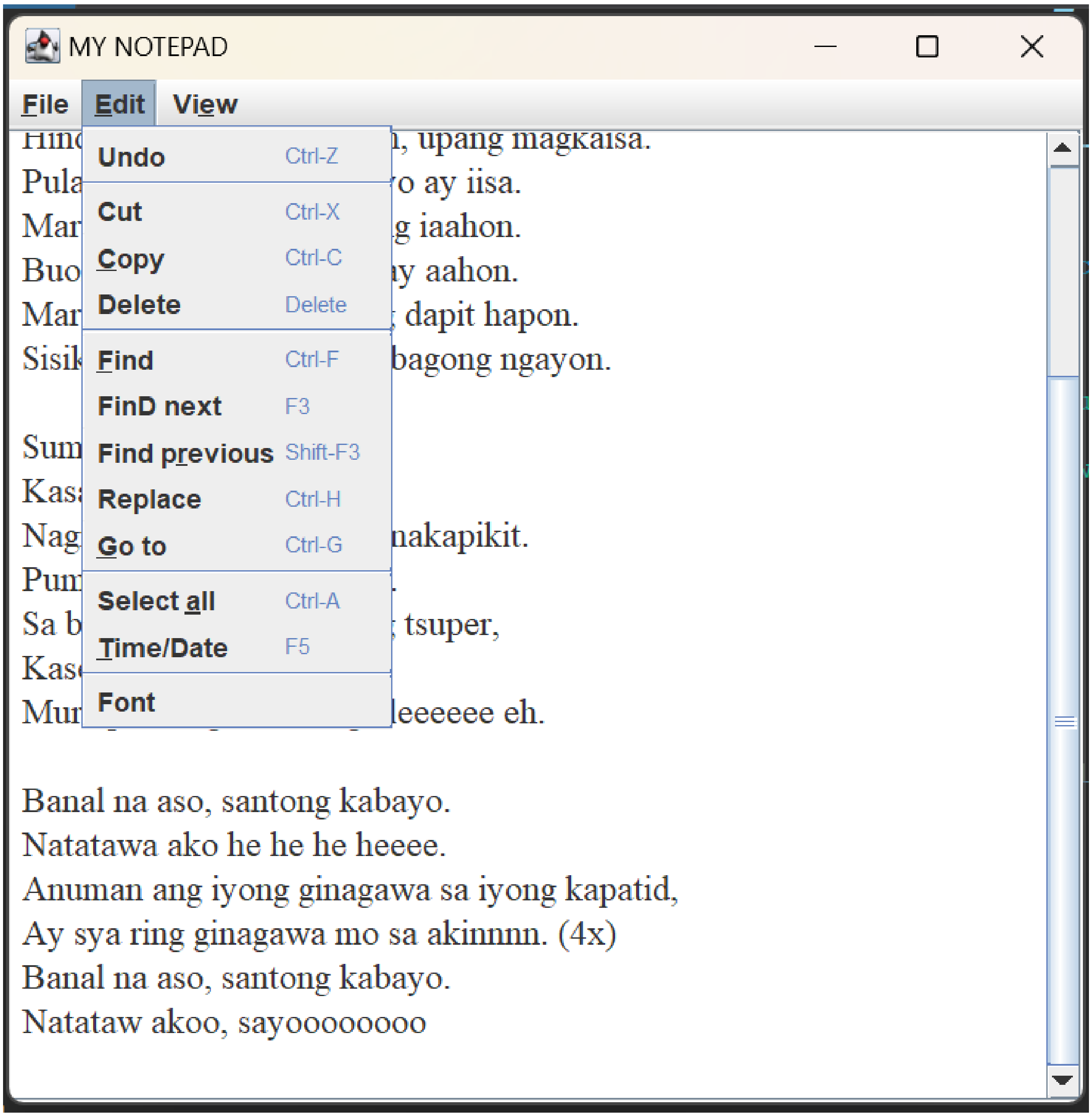


pusot diwa.
ating ibabangon.
yo ay aahon.
g isang dapit hapon.
ng bagong ngayon.

gan, upang magkaisa.
tayo ay iisa.
Marawi sabay sabay nating iaahon.
Buong pagmamahal tayo ay aahon.
Marawi darating ang isang dapit hapon.
Sisikat ang araw, sa isang bagong ngayon.

Sumakay ako sa jeepney,
Kasabay ang isang ale.
Nagrorosaryo, mata nyay nakapikit.
Pumara sa may kumbento.
Sa babaan lang pio sabi ng tsuper,
Kase may nanghuhuli.
Mura parin ng mura ang aleeeeee eh.

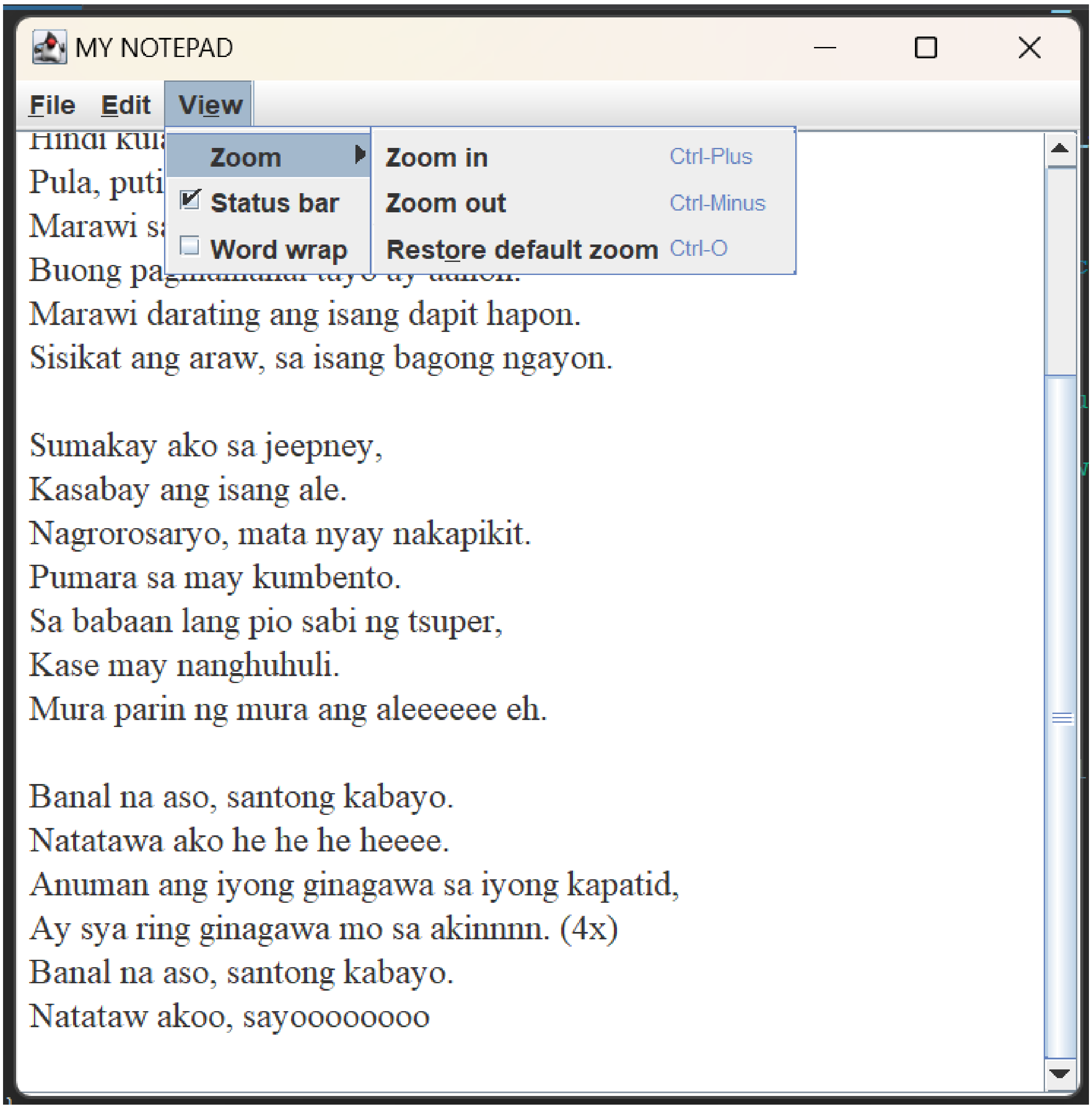
Banal na aso, santong kabayo



Hinc n, upang magkaisa.
Pula o ay iisa.
Mar g iaahon.
Buo y aahon.
Mar dapit hapon.
Sisik bagong ngayon.

Sum
Kas
Nag nakapikit.
Pun .
Sa b tsuper,
Kas
Mur leeeeee eh.

Banal na aso, santong kabayo.
Natatawa ako he he he heeee.
Anuman ang iyong ginagawa sa iyong kapatid,
Ay sya ring ginagawa mo sa akinnnn. (4x)
Banal na aso, santong kabayo.
Natataw akoo, sayoooooooo



Hindi kul
Pula, puti
Marawi s
Buong pa
Marawi darating ang isang dapit hapon.
Sisikat ang araw, sa isang bagong ngayon.

Sumakay ako sa jeepney,
Kasabay ang isang ale.
Nagrorosaryo, mata nyay nakapikit.
Pumara sa may kumbento.
Sa babaan lang pio sabi ng tsuper,
Kase may nanghuhuli.
Mura parin ng mura ang aleeeeee eh.

Banal na aso, santong kabayo.
Natatawa ako he he he heeee.
Anuman ang iyong ginagawa sa iyong kapatid,
Ay sya ring ginagawa mo sa akinnnn. (4x)
Banal na aso, santong kabayo.
Natataw akoo, sayoooooooo

1.4 Questions

1. Is Menubar, Menu and Menuitem useful in creating a GUI? why or why not.

These menu options offer users the option to interact with a program using a GUI, or graphical user input, just like they do in desktop apps and websites. A menu is a Java Swing component that enables developers to arrange items in a container in Java applications. You can have multiple menu items on your menu bar, which is where menus are displayed.

2. Why do Menubar, Menu and Menuitem present in almost all GUI applications?

The majority of GUI apps contain menus. They have a significant role in the simple click of a button with the user's interactions with the program, which may be their single option. Even so, menus are containing areas for widgets that, when clicked, execute operations. Menus contain menu items. Despite the fact that toolbars are more universal than this they are typically used to offer instant access to frequently used menu items, however, they can include any widget.

1.5 Conclusion

In conclusion, in graphical user interfaces, a menu is a type of control graphic. The user can select from a selection of options or commands to carry out the appropriate application function. Operating systems, software programs, and Web-based applications frequently have menus. They aid in improving the user's content's visual presentation, organizing, and classification. The user can access functionality by choosing from a variety of options on a menu. Users find them simpler to use compared to other interfaces. They are well-organized and make it possible to move between the several levels of the structure. Sub-menus can also be used to organize menus. The menu item is frequently chosen by being highlighted using the keyboard, mouse, joystick, or other input devices. There are many different ways to implement menus, including text-based menus, pull-down menus, pop-up menus, context-based menus, and even menus that combine text and symbols. Applications that are menu-driven are typically thought to be more user-friendly. They provide consumers a more natural manner to connect and are more adaptable. The user is given every option available, therefore there is no need for them to memorize any choices.

Output Screenshot: Sample Program and Activity Problem. (1 each menu item)