

1)

Popravi metodo `sortiraj` tako, da bo uredila tabelo nizov `a` glede na njihovo dolžino: na prvem mestu naj bo najdaljši niz, na drugem mestu naslednji po dolžini, itd. Če je več nizov enako dolgih, naj bodo urejeni po abecedi. Popravke metode vnesite kar v spodnjo kodo.

```
public static void sortiraj(int[] a)
{
    int iMin;
    int vMin;

    for(int i = 0; i < a.length - 2; ++i)
    {
        iMin = i;
        vMin = a[i];

        for(int j = i + 1; j < a.length - 1; ++j)
        {
            if(a[j] < vMin)
            {
                iMin = j;
                vMin = a[j];
            }
        }

        a[iMin] = a[i];
        a[i] = vMin;
    }
}
```

Popravljen koda:

```
public class Prva
{
    public static void main(String[] args)
    {
        String[] tabela = {"Anja", "Abja", "Matej", "a", "kbj", "kaj",
"b"};

        sortiraj(tabela);
    }

    public static void sortiraj(String[] a)
    {
        int iMin;
        String vMin;

        for(int i = 0; i < a.length - 2; ++i)
        {
            iMin = i;
            vMin = a[i];

            for(int j = i + 1; j < a.length - 1; ++j)
            {
                if(a[j].length() > vMin.length())
                {
                    iMin = j;
                    vMin = a[j];
                }
            }
        }
    }
}
```

```

        if(a[j].length() == vMin.length())
        {
            if(a[j].compareTo(vMin) < 0)
            {
                iMin = j;
                vMin = a[j];
            }
        }
    }

    a[iMin] = a[i];
    a[i] = vMin;
}

for(int i = 0; i < a.length; i++)
{
    System.out.println(a[i]);
}
}

```

2)

V Javi napiši aplet, ki vsebuje gumb (Jbutton) in ploščo (Jpanel). Ob pritisku na gumb se barva plošče spreminja ciklično med belo, modro, zeleno, rdečo, rumeno in črno. Začetna barva naj bo bela. Barve hranite v posebni tabeli.

```

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

public class Druga extends JApplet implements ActionListener
{
    private JButton gumb = new JButton("Spremeni barvo");
    private JPanel plosca = new JPanel();
    private Container vsebina = getContentPane();
    private Color[] barve = {Color.white, Color.blue, Color.green,
Color.red, Color.yellow, Color.black};
    private int stevec = 0;

    public void init()
    {
        setSize(200, 200);
        setLayout(new BorderLayout());

        vsebina.add(gumb, BorderLayout.NORTH);
        vsebina.add(plosca, BorderLayout.SOUTH);

        gumb.addActionListener(this);
        plosca.setBackground(barve[stevec++]);
    }

    public void actionPerformed(ActionEvent e)
    {
        Object izvor = e.getSource();

        if(izvor == gumb)
        {
            if(stevec > barve.length - 1)
            {

```

```
        stevec = 0;
    }
    plosca.setBackground(barve[stevec++]);
}
}
```

3)

Imaš linearni seznam, ki hrani celoštevilске elemente. Napiši funkcijo, ki iz njega izloči element. Funkcija ima dva parametra: kazalec na začetek seznama in kazalec na element, ki ga hočemo izločiti. Definirajte tudi strukturo, ki določa elemente seznama. Pri brisanju ne pozabite sprostiti pomnilnik.

```
#include <stdio.h>

struct element
{
    int stevilo;
    struct element *naslednji;
};

//ne dela, tako da prosim za pomoc (!!!!) - kazalci niso ravno moja strast
//pa me zanima kje sem ga kiksnila
void izloci(struct element *zacetek, *izlociE)
{
    struct element *t;
    struct element *p;

    t = (struct element*)malloc(sizeof(struct element));

    zacetek = NULL;
    zacetek = zacetek->naslednji;

    while(zacetek != NULL)
    {
        if(izlociE == t->stevilo)
        {
            p = t;
            t = t->naslednji;
            free(p);
        }
        else
        {
            t = t->naslednji;
        }
    }
}
```

4)

Napišite funkcijo, ki v datoteki poišče in na standardni izhod izpiše vse besede, ki se začnejo na določeno črko. Ime datoteke in črka naj bosta podana kot parametra funkcije.

```

#include <stdio.h>
#include <string.h>
#include <ctype.h>

void poisci(char *imeD, char crka)
{
    FILE *dat;
    dat = fopen(imeD, "r");
    char vrsticaP[100];
    char vrstica[100] = " ";
    char c[1] = " ";
    int i;
    int j;

    while(fgets(vrsticaP,100,dat) != NULL)
    {
        strcat(vrstica, vrsticaP);
        strcat(vrstica, c);

        for(i=0; i<strlen(vrstica)-1; i++)
        {
            if(isspace(vrstica[i]) && (vrstica[i+1] == crka))
            {
                printf("%c", vrstica[i+1]);

                for(j=i+2; j<strlen(vrstica)-1; j++)
                {
                    if(!isspace(vrstica[j]))
                    {
                        printf("%c", vrstica[j]);
                    }
                    else
                    {
                        i = j - 1;
                        printf("\n");
                        break;
                    }
                }
            }
        }

        strcpy(vrstica, " ");
    }

    fclose(dat);
}

main()
{
    poisci("1.txt", 'a');
}

```