



**UNIWERSYTET TECHNOLOGICZNO-PRZYRODNICZY
IM. J. I J. ŚNIADECKICH W BYDGOSZCZY**

WYDZIAŁ TELEKOMUNIKACJI, INFORMATYKI I ELEKTROTECHNIKI

PROGRAMOWANIE APLIKACJI MOBILNYCH

ĆWICZENIA Z ZAKRESU TWORZENIE APLIKACJI TYPU MIDLET LABORATORIUM 1

AUTOR:

ŁUKASZ KRĘSKI

DATA WYKONANIA

13.11.2017

DATA ODDANIA

20.11.2017

KIERUNEK:

INFORMATYKA STOSOWANA

GRUPA: 4

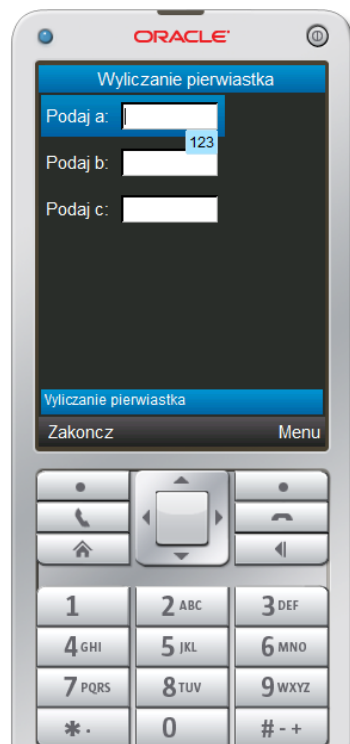
SEMESTR: V

ROK AKADEMICKI: 2017/18

TRYB STUDIÓW: STACJONARNE

Zadanie 1

Utworzyć Midlet wczytujący współczynniki równania kwadratowego $ax^2 + bx + c = 0$ i wyświetlić rozwiązanie równania (w dziedzinie liczb rzeczywistych).



Rysunek 1: Wygląd interfejsu użytkownika

```
public class Midlet extends MIDlet implements CommandListener{
    private Form okno;
    private TextField wspolczynnikA;
    private TextField wspolczynnikB;
    private TextField wspolczynnikC;
    private Command cmdPrzelicz;
    private Ticker komunikat;
    private Command cmdKoniec;

    /*
    Zostaje stworzona klasa Midlet, rozszerza ona Klasę MIDlet oraz implementuje
    CommandListnera.
    Deklarujemy jako zmienne prywatne elementy, które będą widoczne w interfejsie
    użytkownika.
        • private Form okno - deklarujemy obiekt klasy Form. Do tej zmiennej zostają
        dopisane wszystkie elementy podklasy Item. Pozwala na dodanie min. Pół
        tekstowych.
        • private TextField (wspolczynnikA, wspolczynnikB, wspolczynnikC) - jest to
        edytowalne pole tekstowe, które możemy umieścić w Form'ie (okno). Jest
        podklasą klasy Item
        • private Command (cmdPrzelicz, cmdKoniec) - rozszerza klasę obiekt. Zachowuje
        informacje o stanie komendy, nie sam proces który się dzieje po jej
        uruchomieniu
        • private Ticker - pozwala na animowanie przewijanego tekstu u dołu ekranu.

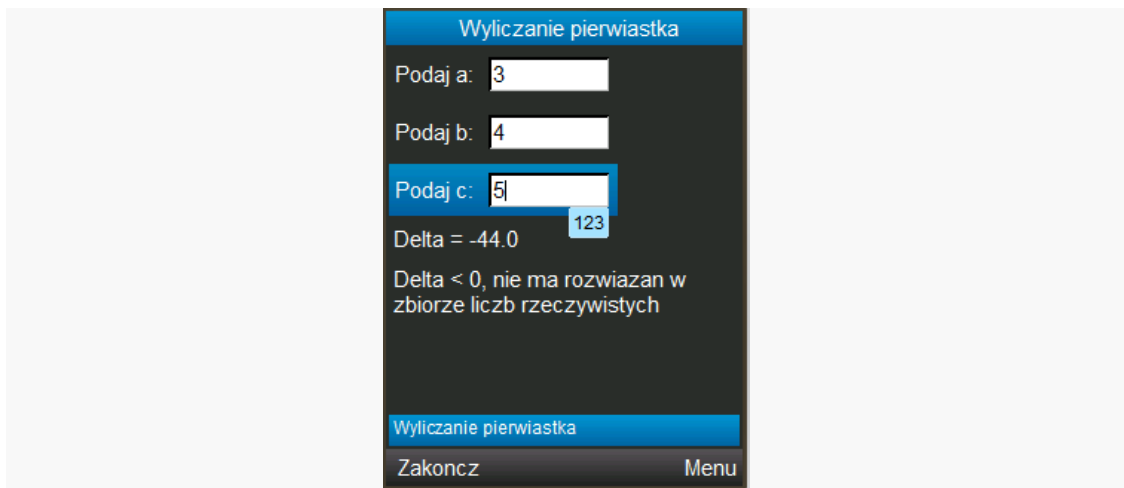
    */

    public Midlet() {
        okno = new Form("Wyliczanie pierwiastka");
        wspolczynnikA = new TextField("Podaj a: ", null, 5, TextField.DECIMAL);
        wspolczynnikB = new TextField("Podaj b: ", null, 5, TextField.DECIMAL);
        wspolczynnikC = new TextField("Podaj c: ", null, 5, TextField.DECIMAL);
        /* Powołanie obiektu Form (okno) z tytułem „Wyliczanie pierwiastka”. Oraz utworzenie
        obiektów klasy TextField. W konstruktorze podawane są następujące parametry: tekst
        znajdujący się przy polu do edycji, wartość ustawiona przez programistę w polu
        edycji, maksymalna długość wartości pola w liczbie znaków (gdy wartość ta = 5, max
```

```

        liczba to 99999, specyfikacja pola (może być np. password wtedy pole to jest
        haszowane)
    */
    komunikat = new Ticker("Wyliczanie pierwiastka");
    cmdPrzelicz = new Command("Przelicz", Command.ITEM, 0);
    cmdKoniec = new Command("Zakonc", Command.EXIT, 0);
    okno.append(wspolczynnikA);
    okno.append(wspolczynnikB);
    okno.append(wspolczynnikC);
    okno.setTicker(komunikat);
    /*Tworzenie nowego obiektu Tickera z w/w tekstem oraz obiektów Command, dodanie
    obiektów typu TextField oraz Tickera do okna. */
    }
    public void startApp() {
        Display ekran = Display.getDisplay(this);
        ekran.setCurrent(okno);
        okno.addCommand(cmdKoniec);
        okno.addCommand(cmdPrzelicz);
        okno.setCommandListener(this);
    }
    /* W tej metodzie ustalamy co ma się dzieć po uruchomieniu aplikacji. W tym
    przypadku ustawiamy okno które ma zostać wyświetlone oraz dodajemy Akcje, które mogą
    zostać wykonane oraz ich nasłuchiwanie */
    }
    public void commandAction(Command c, Displayable d) {
        if(c==cmdPrzelicz){
            try{
                double a= Double.parseDouble(wspolczynnikA.getString());
                double b= Double.parseDouble(wspolczynnikB.getString());
                double wc= Double.parseDouble(wspolczynnikC.getString());
                double delta = (b*b)-(4*a*wc);
                if(a==0 && b==0 && wc==0){
                    okno.append("Funkcja ma nieskonczenie wiele rozwiazan");
                }
                else if (a==0){
                    okno.append("Nie jest to funkcja kwadratowa");
                }
                else {
                    if(delta==0){
                        double mianownik= 2*a;
                        double x0= -b/mianownik;
                        okno.append("Delta = "+delta);
                        okno.append("x0= " + x0); }
                    if (delta>0){
                        double x1=(-b+Math.sqrt(delta))/(2*a);
                        double x2=(-b-Math.sqrt(delta))/(2*a);
                        okno.append("x1 = " +x1);
                        okno.append("x2= " +x2); }
                    if(delta<0){
                        okno.append("Delta = " + delta);
                        okno.append("Delta < 0, nie ma rozwiazan w zbiorze liczb rzeczywistych");
                    } }
                }
            } catch(Exception e){}
        }
        if(c==cmdKoniec){
            notifyDestroyed();
        }
    }
    /* W tej metodzie definiujemy co ma się dzieć po wykonaniu konkretnej akcji. W
    zadaniu pierwszy należało wyliczyć pierwiastki funkcji kwadratowej, jeżeli
    akcja==cmdPrzelicz pobierane są wartości z TextFieldów i na ich podstawie wykonywane
    są obliczenia, jeżeli akcja=cmdKoniec to odnotowujemy zamknięcie aplikacji oraz
    zabijamy proces */

```



Zadanie 2

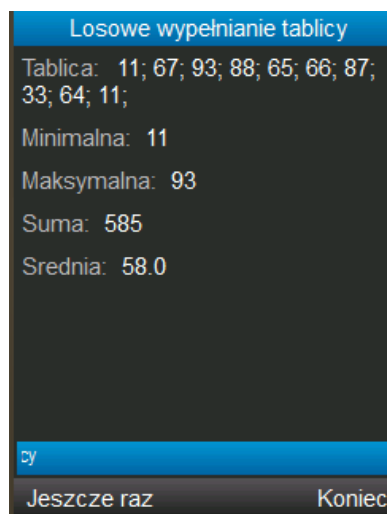
Utworzyć Midlet wypełniający tablicę jednowymiarową losowymi wartościami i wyznaczający: wartość najmniejszą, wartość największą, sumę, średnią arytmetyczną.

```
public class Midlet extends MIDlet implements CommandListener {
    private Form okno;
    private Ticker komunikat;
    private Command cmdJeszczeRaz;
    private Command cmdKoniec;
    private StringItem TablicaItem, MinItem, MaxItem, SumaItem, SredniaItem;
    public Midlet() {
        TablicaItem = new StringItem("Tablica: ", "");
        MinItem = new StringItem("Minimalna: ", "");
        MaxItem = new StringItem("Maksymalna: ", "");
        SumaItem = new StringItem("Suma: ", "");
        SredniaItem = new StringItem("Średnia: ", "");
        okno = new Form("Losowe wypełnianie tablicy");
        okno.append(TablicaItem);
        okno.append(MinItem);
        okno.append(MaxItem);
        okno.append(SumaItem);
        okno.append(SredniaItem);
        komunikat = new Ticker("Losowe wypełnianie tablicy");
        cmdJeszczeRaz = new Command("Jeszcze raz", Command.ITEM, 1);
        cmdKoniec = new Command("Koniec", Command.ITEM, 0);
        okno.setTicker(komunikat);
    }
    public void startApp() {
        //uruchamianie przy starcie aplikacji
        Display ekran = Display.getDisplay(this);
        ekran.setCurrent(okno);
        okno.addCommand(cmdJeszczeRaz);
        okno.addCommand(cmdKoniec);
        okno.setCommandListener(this);
        obliczanie();
    }
    public void obliczanie() {
        //ustawienia programu
        int liczbaelementow=10;
        int liczbalo=100;
        //generator liczb pseudolosowych
        int[] tablo = new int[liczbaelementow];
        Random generator = new Random();
        //zmienne pomocnicze
        int min=100;
        int max=0;
```

```

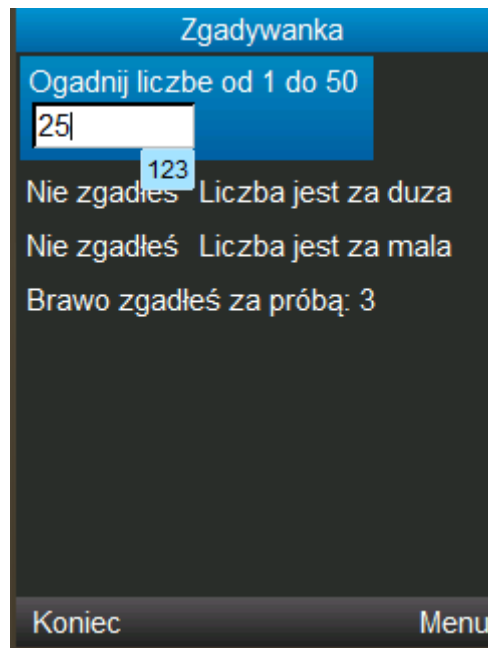
int suma=0;
String los="";
//losowanie liczb
for (int i=0; i<=liczbaelementow-1;i++)
{
    tablos[i]=generator.nextInt(liczbalos);
    if (tablos[i]>max)
    {
        max=tablos[i];
    }
    if (tablos[i]<min)
    {
        min=tablos[i];
    }
    suma+=tablos[i];
    los=los+ tablos[i]+" ";
    TablicaItem.setText(los);
}
//wyświetlanie wyniku
MinItem.setText(Integer.toString(min));
MaxItem.setText(Integer.toString(max));
SumaItem.setText(Integer.toString(suma));
SredniaItem.setText(Float.toString(suma/liczbaelementow));
}
public void commandAction(Command cmd, Displayable d) {
    if (cmd == cmdKoniec) {
        //zamknięcie programu
        notifyDestroyed(); }
    if (cmd == cmdJeszczeRaz) {
        obliczanie();}}

```



Zadanie 3

Utworzyć Midlet zgadywanek: losowana jest liczba naturalna z pewnego zakresu, który jest znany użytkownikowi. Zadaniem użytkownika jest odgadnięcie tej liczby. Po podaniu liczby przez użytkownika Midlet udziela odpowiedzi: "za mała", "za duża" lub "trafiłeś za n-tym razem" (gdzie n jest liczbą wykorzystanych prób)



```
public class Midlet extends MIDlet implements CommandListener{
    private TextField zgadywana;
    private Command cmdKoniec;
    private Command cmdSprawdz;
    private Form okno;
    private int liczba;
    private int proba=1;
    /* Analogicznie jak w zadaniu pierwszym. Dodana została zmienna typu int, zliczająca
    ilość prób */
    public Midlet() {
        okno = new Form("Zgadywanka");
        zgadywana = new TextField("Ogadnij liczbę od 1 do
50", null, 5, TextField.DECIMAL);
        okno.append(zgadywana);
        cmdKoniec = new Command("Koniec", Command.EXIT, 0);
        cmdSprawdz = new Command("Sprawdz", Command.ITEM, 0); }
    public void startApp() {
        Display display = Display.getDisplay(this);
        display.setCurrent(okno);
        okno.addCommand(cmdKoniec);
        okno.addCommand(cmdSprawdz);
        okno.setCommandListener(this);
        Random r = new Random();
        liczba = r.nextInt(51);
        System.out.println(liczba); }
    /* Na start aplikacji generujemy liczbę pseudolosową */
    public void commandAction(Command c, Displayable d) {
        if(c==cmdSprawdz){
            String strzal = zgadywana.getString();
            int strzalCyfra = Integer.parseInt(strzal);
            if (strzalCyfra==liczba){
                okno.append("Brawo zgadłeś za próbą: " + proba);}
            else{
                okno.append("Nie zgadłeś");
                proba++;
                if (strzalCyfra<liczba) okno.append("Liczba jest za mała");
                if (strzalCyfra>liczba) okno.append("Liczba jest za duża");
            } } } }
```

Wnioski

Na zajęciach zapoznaliśmy się z podstawowymi zagadnieniami tworzenia interfejsu użytkownika oraz nadawania akcji poszczególnym zdarzeniom nasłuchiwanym. Tworzenie UI w J2ME w znacznym stopniu przypomina tworzenie UI przy pomocy biblioteki AWT/SWING w Javie SE.