

Esercizio 1

Corretto

```
public class Rettangolo{

    private int base;
    private int altezza;

    Rettangolo()
    {
        base = 1;
        altezza = 1;
    }
    Rettangolo(int base, int altezza)
    {
        this.base = base;
        this.altezza = altezza;
    }

    public int getArea()
    {
        return base*altezza;
    }
    public int getPerimetro()
    {
        return (base+altezza)*2;
    }
    public void setAltezza(int altezza)
    {
        this.altezza = altezza;
    }
    public void setBase(int base)
    {
        this.base = base;
    }
}
```

Esercizio 2 corretto

```
import java.util.Random;
public class Permutazioni
{
    private int[] v;
    Permutazioni()
    {
        v = new int[10];
    }

    public void visualizza()
    {
        for(int i = 0; i < 10; i++)
        {
            System.out.print(v[i] + " ");
        }
        System.out.println();
    }

    public void carica()
    {
        for(int i = 0; i < 10; i++)
        {
            v[i] = 0;
        }
        Random r = new Random();
        boolean flag;
```

```

        for(int i = 0; i < 10; i++)
        {
            do{
                flag = false;
                v[i] = r.nextInt(10)+1;
                for(int j = 0; j < i; j++)
                {
                    if(v[i] == v[j])
                    {
                        flag = true;
                    }
                }
            }while(flag==true);
        }
    }
}

public class MainPermutazioni
{
    public static void main(String[] args)
    {
        Permutazioni p = new Permutazioni();

        for(int i = 0; i < 10; i++)
        {
            p.carica();
            p.visualizza();
        }
    }
}

```

Esercizio 3 metodo getLengV3 sbagliato e merge non perfettamente funzionante

```

public class Fusione
{
    private int[] v1;
    private int[] v2;
    private int[] v3;

    Fusione(int[] v1, int[] v2)
    {
        this.v1 = new int[v1.length];
        for(int i = 0; i < v1.length; i++)
            this.v1[i] = v1[i];
        this.v2 = new int[v2.length];
        for(int i = 0; i < v2.length; i++)
            this.v2[i] = v2[i];
        int dim = getLengV3();
        this.v3 = new int[dim];
    }

    // praticamente rifatto
    private int getLengV3()
    {
        int i = 0;
        int j = 0;
        int k = 0;
        while(i < v1.length && j < v2.length)
        {
            if(v1[i] < v2[j])
            {
                i++;
                k++;
            }
        }
    }
}

```

```

        } else if(v1[i] > v2[j]) {
            j++;
            k++;
        } else {
            i++;
            j++;
            k++;
        }
    }
}
while(i < v1.length)
{
    i++;
    k++;
}
while(j < v2.length)
{
    j++;
    k++;
}
return k;
}

```

```

public void visualizzaV1()
{
    for(int i = 0; i < v1.length; i++)
        System.out.print(v1[i] + " ");
    System.out.println();
}
public void visualizzaV2()
{
    for(int i = 0; i < v2.length; i++)
        System.out.print(v2[i] + " ");
    System.out.println();
}
public void visualizzaV3()
{
    for(int i = 0; i < v3.length; i++)
        System.out.print(v3[i] + " ");
    System.out.println();
}
// leggermente cambiato
public void merge()
{
    int i = 0;
    int j = 0;
    int k = 0;
    while(i < v1.length && j < v2.length)
    {
        if(v1[i] < v2[j])
        {
            v3[k] = v1[i];
            i++;
            k++;
        } else if(v1[i] > v2[j]) {
            v3[k] = v2[j];
            j++;
            k++;
        } else {
            v3[k] = v1[i];
            i++;
            j++;
            k++;
        }
    }
}

```

```

    }
}
while(i < v1.length)
{
    v3[k] = v1[i];
    i++;
    k++;
}
while(j < v2.length)
{
    v3[k] = v2[j];
    j++;
    k++;
}
}
}

```

Esercizio 4 corretto

**public class** ContoCorrente

```

{

    private int soldi;
    private int[][] lista;
    ContoCorrente()
    {
        lista = new int[5][2];
        for(int i = 0; i < 5; i++)
        {
            lista[i][1] = -1;
            lista[i][0] = 0;
        }
        soldi = 0;
    }
    public int getSoldi()
    {
        return soldi;
    }
    public void versamento(int n)
    {
        soldi = soldi + n;
        shift(n,1);
    }
    public void prelievo(int n)
    {
        soldi = soldi - n;
        shift(n,2);
    }
    public void shift(int soldi,int tipo)
    {
        for(int i = 3; i >= 0; i--)
        {
            lista[i+1][0] = lista[i][0];
            lista[i+1][1] = lista[i][1];
        }
        lista[0][0] = soldi;
        lista[0][1] = tipo;
    }
    public void listaMovimenti()
    {
        for(int i = 4; i >= 0; i--)
        {

```

```

        if(lista[i][1] == 1)
        {
            System.out.println("Versamento: " + lista[i][0]);
        } else if(lista[i][1] == 2) {
            System.out.println("Prelievo: -" + lista[i][0]);
        }
    }
    System.out.println();
}

}

import java.util.Scanner;
public class MainContoCorrente
{
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        ContoCorrente cc = new ContoCorrente();
        boolean flag = true;

        while(flag)
        {
            System.out.println("v-versamento\tp-prelievo\tl-lista movimenti\tts-
saldo\ttx-exit");
            char scelta;
            try{
                String dummy = input.nextLine();
                scelta = dummy.charAt(0);
            } catch(Exception e)
            {
                System.out.print("Inserire un carattere");
                return;
            }
            int somma;
            switch(scelta)
            {
                case 'v':
                    System.out.print("Somma da versare: ");
                    try{
                        somma = input.nextInt();
                    } catch(Exception e)
                    {
                        System.out.print("Inserire un numero");
                        return;
                    }
                    input.nextLine();
                    cc.versamento(somma);
                    break;
                case 'p':
                    System.out.print("Somma da prelevare: ");
                    try{
                        somma = input.nextInt();
                    } catch(Exception e)
                    {
                        System.out.print("Inserire un numero");
                        return;
                    }
                    input.nextLine();
                    cc.prelievo(somma);
                    break;
                case 'l':
                    cc.listaMovimenti();
                    break;
            }
        }
    }
}

```

```
case 's':  
    System.out.println(cc.getSoldi());  
    break;  
case 'x':  
    flag = false;  
    break;  
default:  
    System.out.println("Scenlta non valida");
```

```
}
```

```
}
```

```
}
```

```
}
```