

---

## ECLP– Lista 2

### 03/03/20

---

1. Suponha as seguintes declarações em Java: `float f1, f2; int i1, i2; short s1, s2;`  
Para cada uma das instruções a seguir determine se:

- é legal e não acontece uma transformação de tipos
- é legal e provoca uma transformação de tipos
- é um erro de compilação
- provoca um erro na execução

a) `f1 = s1;`    b) `s1 = f1;`    c) `i1 = s1;`    d) `s1 = i1;`

2. Quais das seguintes expressões de C são *l-values* levando em consideração a seguinte declaração da função:

```
void f(int x, int* y) { ... }
```

A) `x + 2`    B) `&x`    C) `*&x`    D) `&x + 2`    E) `*(&x + 2)`

Justificar quais são e quais não.

3. a) Quais são as principais componentes da memória na execução de um programa?  
b) Descreva como elas são usadas. Exemplifique em C e Java.
4. Dados os trechos de código a seguir, determinar, segundo o mapa de memória visto nas aulas, onde são armazenadas cada uma das variáveis.

a) 

```
#include <stdio.h>
int main(void)
{
    int n[] = {23, 45, 64};
    int *p;
    p = &n[0]; p++;
    printf("Valor: %d ", *p);
    (*p)++;
    printf("Valor: %d\n", *p);
}
```

b) 

```
program main;
var N, C, I: integer;
Begin
    Readln(N);
    C := 0; I := 1;
    While ( I <= N ) do
        Begin
            Readln(A);
            If (A mod 3=2) then C:= c + 1;
            Inc(I);
        end;
    End;
```

c) 

```
public class RegAcademicoGrad {
    private String nomeA;
```

```

private Data dataN;
private int ra;
private byte curso;
private double perc; // desconto matrícula
public static final byte CC =1;
public static final byte SI =2;
public static final byte WEB =3;
// métodos de RegAcademicoGrad
RegAcademicoGrad(String n, Data d, int m, byte c, double p) {
    nomeA = n; dataN = d; ra = m; curso= c; perc= p; }
RegAcademicoGrad(String n, Data d, int m, byte c) {
    this(n,d,m,c,100) //100% desconto }
public double calcMensa() {
    double mensa = 0;
    if ( curso == WEB) mensa = 550;
    else mensa = 650;
    mensa = (mensa * perc )/100;
    return mensa; }
public String toString() {
    String result =""; result += "Matricula: "+ra+"Nome: "+nomeA+"\n";
    result += "Data de Nascimento:"+ dataN+ "+"+"\n";
    ...
    return result; }
public class TesteRegAcademicoGrad {
    public static void main(String args[]) {
        Data dataNJose=new Data((byte)12,(byte)5,(short)1992);
        RegAcademicoGrad jose = new RegAcademicoGrad("Jose Silva",
        dataNJose, 17345, RegAcademicoGrad.CC, 45);
        RegAcademicoGrad luis = new RegAcademicoGrad("Luis Dias", new
        Data ((byte)25,(byte)8,(short) 1990), 18005, RegAcademicoGrad.CC);
        System.out.println(jose);
        System.out.println(luis);
    }
}

```

5. Considere os seguintes fragmentos de código em Java. Em cada caso determine se o código compila corretamente. Se compila, determinar se executa sem erros ou pode acontecer uma *exception*.

```

String[] strings = new String[10];
Object[] objects;
String s;
objects = strings;
strings[0] = "hello sailor"; s = (String)objects[0];

```

```

String[] strings = new String[10];
Object[] objects;
objects = strings;
objects[0] = new Integer(100);

```

```

String[] strings;
Object[] objects = new Object[10];
String s;
strings = objects;
objects[0] = "hello sailor";
s = strings[0];

```