UTFPR - Universidade Tecnológica Federal do Paraná
DACOM - Departamento de Computação
BCC35A :: Linguagens de Programação

Prova 2

Instruções

- Escreva as respostas com caneta.
- Não é permitido realizar qualquer tipo de consulta.

1) (1,5) Explique e diferencie sobrecarga e sobrescrita de métodos.

2) (1,5) Observe o código C++ abaixo e responda as questões.

```
class Entity {
protected:
    long id; static int count;
public:
    Entity(long id) {
       this->id = id;
       Entity::count++;
    }
    virtual void display() {
       cout << "\n" << this->id << " ";
    }
};
// inicia campo estático
int Entity::count = 0;
```

```
class Person : public Entity {
private:
    string name;
public:
    Person(long id, string name):
                              Entity(id) {
        this->name = name;
    }
    void display() {
        Entity::display();
        cout << this->name;
        cout << " #" << Entity::count;
    }
};
```

a) Qual a saída impressa pelo seguinte código?

```
Entity* list[2];
list[0] = new Person(10, "Adam Jensen");
list[1] = new Entity(20);
for (Entity* e : list)
    e->display();
```

b) Para que serve a palavra virtual no método Entity::display()? O que ocorre se não for usada?

c) Defina campo estático. Defina método estático. Quando eles podem ser chamados. Qual acesso as instâncias da classe podem ter sobre campos e métodos estáticos.

3) (1,5) Escreva uma função LISP que recebe uma lista de números e devolve a quantidade de valores positivos que estão na lista.

4) (1,5) Escreva uma função **recursiva** LISP que imprime os números pares de N até 2. A função recebe N como parâmetro.

5) (1,5) Observe o código C# abaixo e responda as questões.

a) Qual a saída do programa quando a chamada da linha 11 é feita com division(25, 5)?
b) Qual a saída do programa quando a chamada da linha 11 é feita com division(0, 5)?
c) Qual a saída do programa quando a chamada da linha 11 é feita com division(0, 0)?

```
01   public class Program {
02        public float division(int num1, int num2) {
03            float res = 0;
04            res = num1 / num2;
05            if (num1 < 0) throw new Exception("We don't like negative numbers!");
06            return res;
07        }
08        public static void Main(string[] args) {
09            float res = 0;
10            try {
11                res = new Program().division(?, ?);
12            } catch (DivideByZeroException e) {
13                Console.WriteLine("Division By Zero");
14            } catch (Exception e) {
15                Console.WriteLine(e.Message);
16            } finally {
17                Console.WriteLine("Result: {0}", res);
18            }
19        }
20    }
```

6) (2,5) Considere a seguinte base de fatos em Prolog:

```
trabalha(joao, microsoft).
trabalha(paula, microsoft).
trabalha(carla, microsoft).
trabalha(marcio, microsoft).
trabalha(jonathan, microsoft).
trabalha(regina, apple).
```

```
funcao(joao, engenheiro).
funcao(paula, engenheiro).
funcao(carla, suporte).
funcao(jonathan, testes).
funcao(regine, engenheiro).

chefe(jonathan, suporte).
chefe(marcio, engenheiro).
```

a) Escreva as seguintes pesquisas:
   - Marcio é chefe?
   - Quem é engenheiro?
   - Lista de pessoas e funções
   - Quem trabalha na microsoft e é engenheiro?

b) Escreva as seguintes regras:
   - Duas pessoas são colegas de trabalho se trabalham na mesma empresa.
   - Uma pessoa trabalha com desenvolvimento em uma empresa se ela trabalha na empresa e tem como função 'engenheiro' ou 'testes'. OBS: você pode usar parênteses para expressões lógicas. Ex: A , (B ; C)
   - Uma pessoa é chefe de outra se ambas trabalham na mesma empresa e a primeira é chefe da função da segunda