

## 1 Primeira questão (8 pontos)

Considere um triângulo de Pascal de altura  $n$  qualquer (a primeira linha está na altura zero).

1. Para exibir um triângulo de Pascal, com a forma correta, quantos espaços em branco devem ser colocados no início da linha na altura  $i$ , sabendo que a altura máxima é *level*?  
 $level - i + 1$
2. Complete os procedimentos, do programa Pascal\_Triangle, para que ele produza a saída desejada:  
*binomial\_coefficient*: retorna o  $m$ -ésimo coeficiente da linha  $n$  do triângulo de Pascal.  
*binomial\_expansion*: expande as potências do polinômio  $(x + y)^e$ . Retorna a string 'bstr' com o resultado da expansão.
3. Complete os dois comandos do programa principal que estão marcados com '...'
4. Desenhe uma interface para o Lazarus que permita fornecer o último nível, exibir a expansão de um polinômio  $(x + y)^n$  e imprimir (ou desenhar) os elementos do triângulo de Pascal. Forneça os nomes de todos os componentes utilizados.

## 2 Segunda questão (2 pontos)

Nesta questão, considere uma aplicação que apresenta uma lista de locais com fotos populares recentes do Flickr (Yahoo), e permite a um usuário visualizar fotos tiradas destes locais. Por exemplo, imagine que a cidade de Londres é um destes locais e que há 50 fotos dos dois últimos dias.

Suponha que a aplicação acessa o site do Flickr, recebe a lista de locais e, para cada local obtém a lista de fotos dos dois últimos dias, e monta um banco de dados de sua escolha. Para cada foto é armazenada apenas a URL do Flickr de onde a foto pode ser baixada.

1. Desenhe o formulário da aplicação e indique os componentes utilizados.
2. Qual a estrutura (colunas) das tabelas SQL do seu banco de dados?

Na Figura 1 pode ser visto o diagrama entidade-relacionamento de um banco de dados adequado a esta questão. Claramente, há três entidades: Região (local), Foto e Fotógrafo (não pedido). Cada uma dessas entidades possui um conjunto de atributos próprio, e no mínimo o identificador da foto, sua URL, data de criação e o nome da região devem estar presentes. Cada foto pertence a uma única região, mas cada região pode estar associada a várias fotos.

Qualquer resposta que faça algum sentido, baseado no que foi descrito acima, será aceita.

```
Program Pascal_Triangle;
```

```
Type
```

```
    ArrayType = Array of longint;
```

```
Var
```

```
    i, j, level: Integer;
```

```
    s: AnsiString;
```

```
Const
```

```
    MAXN = 20;
```

```
{ computa a combinacao de n, m a m }
```

```
function binomial_coefficient (n, m: Integer): longint;
```

```
    var i, j: Integer; { contadores }
```

```
    bc: Array [0..MAXN] of ArrayType; { tabela de coeficientes binomiais }
```

```
begin
```

```
    if (n > MAXN) or (m > n) then begin
```

```
        binomial_coefficient := 0;
```

```
        exit
```

```
    end;
```

```
    for i := 0 to n do begin
```

```
        SetLength(bc[i], n+1);
```

```
        bc[i][0] := 1;
```

```
        bc[i][i] := 1;
```

```
    end;
```

```
    for i := 1 to n do
```

```
        for j := 1 to i-1 do
```

```
            bc[i][j] := bc[i-1][j-1] + bc[i-1][j];
```

```
        binomial_coefficient := bc[n][m];
```

```
    end;
```

```
procedure binomial_expansion ( e: Integer; var bstr: AnsiString );
```

```
    var b, s, sx, sy: String;
```

```
    i: Integer;
```

```
begin
```

```
    if e = 0 then begin
```

```
        bstr := '1';
```

```
        exit;
```

```
    end;
```

```
    str(e, s); {converte o inteiro 'e' para a string 's'}
```

```
    bstr := 'x';
```

```
    if s <> '1' then bstr += '^' + s;
```

```
    bstr += ' + ';
```

```
    for i := 1 to e-1 do begin
```

```
        str(e-i, sx);
```

```
        str(i, sy);
```

```
        str(binomial_coefficient(e, i), b);
```

```
        bstr += b + 'x';
```

```

        if sx <> '1' then bstr += '^' + sx;
        bstr += 'y';
        if sy <> '1' then bstr += '^' + sy;
        bstr += ' + ';
    end;
    bstr += 'y';
    if s <> '1' then bstr += '^' + s;
end;
begin
    level := 5;
    Write ( 'Forneca o ultimo nivel do triangulo de Pascal: ' );
    Read(level);
    Writeln();

    for i := 0 to level do begin
        for j := 0 to level-i+1 do
            write( ' ' );
        for j := 0 to i do
            write ( binomial_coefficient(i,j), ' ' );
        writeln();
    end;
    writeln();

    for i := 0 to level do begin
        binomial_expansion ( i, s );
        writeln ( s )
    end
end.

```

Saida do programa:

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
 1 5 10 10 5 1
1 6 15 20 15 6 1
```

```
1
x + y
x^2 + 2xy + y^2
x^3 + 3x^2y + 3xy^2 + y^3
x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4
x^5 + 5x^4y + 10x^3y^2 + 10x^2y^3 + 5xy^4 + y^5
x^6 + 6x^5y + 15x^4y^2 + 20x^3y^3 + 15x^2y^4 + 6xy^5 + y^6
```

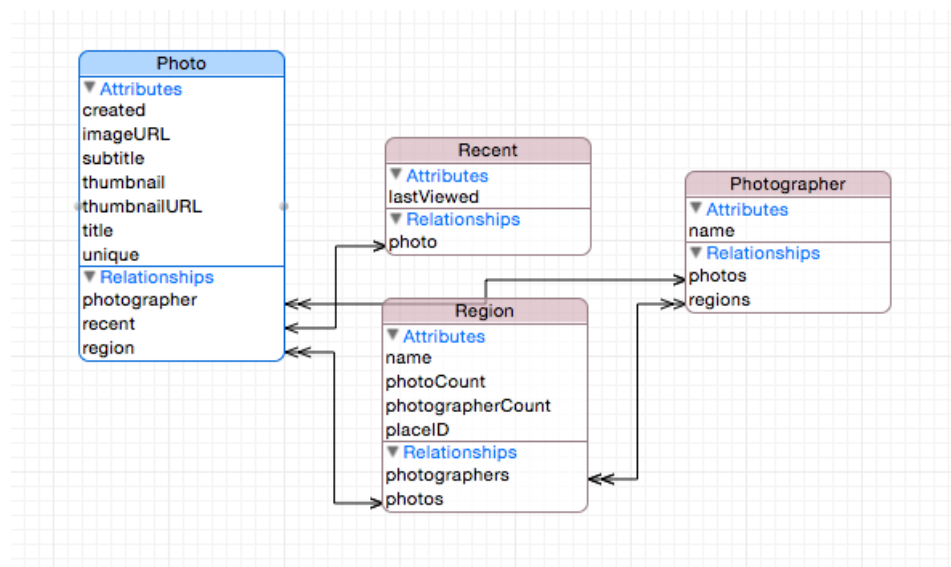


Figura 1: Diagrama Entidade-Relacionamento.