

DCpic (4.3.2)

Manual de Utilização

Pedro Quaresma
CISUC/Departamento de Matemática, Universidade de Coimbra
3001-454 COIMBRA, PORTUGAL
pedro@mat.uc.pt phone: +351-239 791 137 fax: +351-239 832 568

Resumo

O *DCpic* é um conjunto de comandos para a escrita de grafos, para tal desenvolveu-se um conjunto de comandos, com uma sintaxe simples, que permite a construção de quase todo o tipo de grafos.

Originalmente o *DCpic* (**D**igramas **C**omutativos utilizando o **PiCTeX**) foi concebido para a construção de diagramas comutativos tal como são usados em Teoria das Categorias (Herrlich & Strecker, 1973; Pierce, 1998), temos então grafos etiquetados e com elementos nos nós. A partir da versão 4.0 o conjunto de comandos foi alterada de forma a considerar-se também a construção de grafos dirigidos, e grafos não dirigidos. A forma de os especificar recorre à colocação dos diferentes objectos (nós e arestas) num dado referencial ortonormado,

O *DCpic* está baseado no \LaTeX necessitando deste para poder ser usado.

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3 of this license or (at your option) any later version. The latest version of this license is in <http://www.latex-project.org/lppl.txt> and version 1.3 or later is part of all distributions of LaTeX version 2005/12/01 or later. This work has the LPPL maintenance status ‘maintained’.

The Current Maintainer of this work is Pedro Quaresma (pedro@mat.uc.pt).

This work consists of the files `dcpic.sty`.

1 História

11/1990 – versão 1.0

10/1991 – versão 1.1

9/1993 – versão 1.2 argumento “distância entre as extremidades da seta e os objectos” passou a ser opcional; uma nova opção para as “setas” (opção 3).

2/3/1995 — versão 1.3 foi acrescentado o tipo de seta de aplicação (opção 4) a distância da etiqueta à seta respectiva passou a ser fixa (10 unidades de medida).

15/7/1996 (versão 2.1) O comando `mor` passou a ter uma sintaxe distinta. Os parâmetros 5 e 6 passaram a ser a distância entre os objectos e os extremos da seta o parâmetro 7 é o nome do morfismo e os parâmetros 8 e 9, colocação do morfismo e tipo de morfismo passaram a ser opcionais.

5/2001 (versão 3.0) Implementação do comando `cmor` baseado no comando de desenho de curvas quadráticas pelo `PiCTeX`.

11/2001 (versão 3.1) Modificação das pontas das setas de forma a estas ficarem semelhantes às setas (símbolos) dos `TeX`.

1/2002 (versão 3.2) Modificação dos comandos `obj` e `mor` de forma a introduzir a especificação lógica dos morfismos, isto é, passa-se a dizer qual é o objecto de partida e/ou o objecto de chegada em vez de ter de especificar o morfismo em termos de coordenadas. Por outro lado o tamanho das setas passa a ser ajustado automaticamente em relação ao tamanho dos objectos.

5/2002 (versão 4.0) **Versão incompatível com as anteriores.** Modificação dos comandos `begindc` e `obj`. O primeiro passou a ter um argumento (obrigatório) que nos permite especificar o tipo de grafo que estamos a querer especificar:

- `commdiag` (0), para diagramas comutativos;
- `digraph` (1), para grafos orientados;
- `undigraph` (2), para grafos não orientados.

O comando `obj` modificou a sua sintaxe passou a ter um (após a especificação das coordenadas, um argumento opcional, um argumento obrigatório, e um argumento opcional. O primeiro argumento opcional dá-nos a etiqueta que serve como referência para a especificação dos morfismos, na sua ausência usa-se o argumento obrigatório para esse efeito, o argumento obrigatório dá-nos o “conteúdo” do objecto, nos diagramas comutativos é centrado no ponto dado pelas coordenadas sendo o argumento seguinte simplesmente ignorado, nos grafos o “conteúdo” é colocado numa posição a norte, a noroeste, a este, . . . , sendo que a posição concreta é especificada pelo último dos argumentos deste comando, o valor por omissão é o **norte**.

3/2003 (versão 4.1) A pedido de Jon Barker <jeb1@soton.ac.uk> criei um novo tipo de seta, a seta de sobrejecção. Para já a dupla seta só fica bem nas setas horizontais ou verticais.

12/2004 (versão 4.1.1) Nova versão das setas de sobrejecção que corrige completamente os problemas da solução anterior.

3/2007 (versão 4.2) Acrescenta a directiva “`providespackage`”. Acrescenta linhas a ponteadas e a tracejadas.

5/2008 (versão 4.2.1) Apaga alguns contadores para tentar diminuir o excessivo uso dos mesmos por parte do `PiCTeX`.

8/2008 (versão 4.3) Graças a Ruben Debeerst (debeerst@mathematik.uni-kassel.de), acrescentei uma nova “seta” a “`equalline`”. Após isso decidi também acrescentar “`doublearrow`”; the `doublearrow with opposite directions`; the `null arrow`. This last can be used as a simple form of adding new labels.

12/2008 (version 4.3.1) Para evitar conflitos com outros pacotes o comando “id” é internalizado. O comando “dasharrow” é modificado para “dashArrow” para evitar um conflito com o AMSTeX.

12/2009 (version 4.3.2) Para evitar um conflito com o pacote “hyperref” mudou-se o contador “d” para “deuc”, aproveitei e mudei os contadores “x” e “y” para “xO” e “yO”

2 Introdução

O conjunto de comandos *DCpic* é um conjunto de comandos T_EX (Knuth, 1986) dedicado à escrita de diagramas tal como são usados em Teoria das Categorias (Herrlich & Strecker, 1973; Pierce, 1998), assim como de grafos dirigidos e não dirigidos (Harary, 1972).

Pretendeu-se com a sua escrita ter uma forma simples de especificar grafos, fazendo-o através da especificação de um conjunto de “objectos” (nós do grafo) colocados num dado referencial ortonormado, e através de um conjunto de morfismos (arestas) que os são posicionados explicitamente no referido referencial, ou então, a sua posição é dada especificando qual é o seu nó de partida e qual é o seu nó de chegada.

O gráfico em si é construído recorrendo aos comandos gráficos do P_TC_TE_X.

3 Utilização

Antes de mais é necessário carregar os dois conjuntos de comandos acima referidos, no caso de um documento L^AT_EX (Lamport, 1994) isso pode ser feito com o seguinte comando (no preâmbulo).

```
\usepackage{dcpic,pictex}
```

Nos outros formatos ter-se-á de usar um comando equivalente. Após isso os diagramas podem ser escritos através dos comandos disponibilizados pelo *DCpic*. Por exemplo, os comandos:

```
\begin{dc}[20]
\obj(1,4){$A^B$}
\obj(1,1){$C$}
\obj(3,4){$A$}
\obj(3,1){$C\times B$}
\obj(6,4){$A^B\times B$}
\mor{$C$}{$A^B$}{$f$}
\mor{$C\times B$}{$A$}{$\bar{f}$}[\atleft,\dashArrow]
\mor{$A^B\times B$}{$A$}{$ev$}[\atright,\solidarrow]
\mor{$C\times B$}{$A^B\times B$}{$f\times id$}[\atright,\solidarrow]
\end{dc}
```

produzem o seguinte diagrama:

$$\begin{array}{ccc}
 A^B & & A \xleftarrow{ev} A^B \times B \\
 \uparrow f & & \uparrow \bar{f} \\
 C & & C \times B
 \end{array}
 \begin{array}{ccc}
 & & \nearrow f \times id \\
 & & \\
 & &
 \end{array}$$

O meio ambiente `begin{dc}`, `end{dc}` permite-nos construir um grafo por colocação dos objectos num referencial ortonormado tendo a origem em (0,0). As arestas (morfismos) vão ligar pares de nós (objectos) entre si.

4 Comandos Disponíveis

De seguida apresenta-se a descrição dos comandos, a sua sintaxe e a sua funcionalidade. Os argumentos entre parêntesis rectos são opcionais.

`\begin{dc}{#1}[#2]` – entrada no ambiente de escrita de grafos:

`#1` – tipo de grafo

0 \equiv `\commdiag`, diagrama comutativo;

1 \equiv `\digraph`, grafo orientado;

2 \equiv `\undigraph`, grafo não orientado;

3 \equiv `\cdigraph`, grafo orientado, com objectos circunscritos;

4 \equiv `\cundigraph`, grafo não orientado, com objectos circunscritos.

`#2` – factor de escala (opcional)

valor por omissão: 30

`\end{dc}` – saída do meio ambiente para a escrita de grafos.

`\obj{#1,#2}[#3]{#4}[#5]`: comando de colocação dos nós (objectos).

`#1` e `#2` – coordenadas do centro da caixa que vai conter o texto

`#3` – etiqueta para identificar o objecto (opcional)

`#4` – texto (conteúdo do nó)

`#5` – colocação relativa do objecto (opcional)

0 \doteq `\pcent`, centrado

1 \doteq `\north`, norte

2 \doteq `\northeast`, nordeste

3 \doteq `\east`, este

4 \doteq `\southeast`, sudeste

5 \doteq `\south`, sul

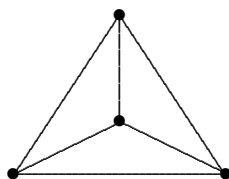
6 \doteq `\southwest`, sudoeste

7 \doteq `\west`, oeste

8 \doteq `\northwest`, noroeste

A etiqueta explícita-se quando não é possível usar o objecto como forma de identificação do nó, por exemplo num dado grafo não orientado os nós podem não ter conteúdo e como tal serem todos iguais em termos de identificação:

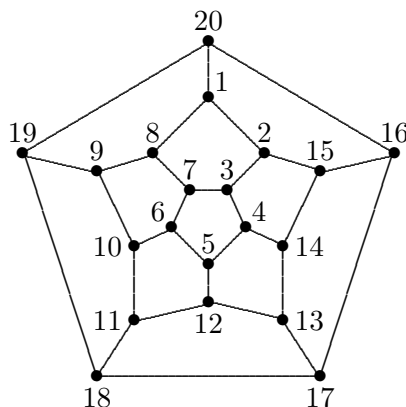
Em alguns casos, por exemplo comandos dos \LaTeX complexos, pode ser necessário explicitar o argumento `#3` mesmo que seja através da etiqueta vazia `{}` . Nesses casos o mecanismo interno do DCpic de comunicação entre comandos baralha-se e entra-se num ciclo infinito.



foi produzido por:

```
\begin{dc}{\undigraph}[20]
\obj(1,1)[1]{}
\obj(3,2)[2]{}
\obj(5,1)[3]{}
\obj(3,4)[4]{}
\mor{1}{2}{}
\mor{1}{3}{}
\mor{2}{3}{}
\mor{4}{1}{}
\mor{4}{3}{}
\mor{2}{4}{}
\end{dc}
```

O parâmetro referente à colocação do objecto só é relevante quando se pensa na identificação dos nós num dado grafo orientado (ou não), por exemplo o grafo “Around the Word” (Harary, 1972):



foi produzido por

```

\begin{dc}\undigraph}[7]
\obj(6,4){18}[\south]
\obj(18,4){17}[\south]
\obj(8,7){11}[\west]
\obj(12,8){12}[\south]
\obj(16,7){13}[\east]
\obj(8,11){10}[\west]
\obj(10,12){6}[\northwest]
\obj(12,10){5}
\obj(14,12){4}[\northeast]
\obj(16,11){14}[\east]
\obj(2,16){19}
\obj(6,15){9}
\obj(9,16){8}
\obj(11,14){7}
\obj(13,14){3}
\obj(15,16){2}
\obj(18,15){15}
\obj(22,16){16}
\obj(12,19){1}[\northeast]
\obj(12,22){20}
\mor{18}{17}{}\mor{18}{19}{}
\mor{11}{12}{}\mor{11}{10}{}\mor{12}{13}{}
\mor{12}{5}{}\mor{10}{6}{}\mor{10}{9}{}
\mor{5}{6}{}\mor{5}{4}{}\mor{13}{17}{}
\mor{13}{14}{}\mor{9}{19}{}\mor{9}{8}{}
\mor{6}{7}{}\mor{4}{3}{}\mor{4}{14}{}
\mor{19}{20}{}\mor{8}{1}{}\mor{8}{7}{}
\mor{7}{3}{}\mor{3}{2}{}\mor{2}{1}{}
\mor{2}{15}{}\mor{14}{15}{}\mor{17}{16}{}
\mor{16}{20}{}\mor{1}{20}{}\mor{15}{16}{}
\end{dc}

```

`\mor{#1}{#2}[#5,#6]{#7}[#8,#9]`: Comando de colocação da seta (morfismo) de ligação de dois objectos – Primeira variante.

A numeração errada dos argumentos é aqui feita propositadamente, aquando da explicação da segunda variante deste comando compreender-se-á o porquê desta opção de escrita.

- #1 – referência do nó de partida
- #2 – referência do nó de chegada

- #5 e #6 – distância do centro dos objectos às extremidades inicial e final respectivamente da seta. Valores por omissão: 10, 10 (para diagramas) 2, 2 (para os grafos)
- #7 – texto, “nome” do morfismo
- #8 – colocação do nome do morfismo em relação à seta. Valor por omissão, `\atleft`.
 - 1 \doteq `\atright`, à direita
 - 1 \doteq `\atleft`, à esquerda
- #9 – tipo da seta. Valor por omissão, `\solidarrow`.
 - 0 \doteq `\solidarrow`, seta sólida
 - 1 \doteq `\dashArrow`, seta tracejada
 - 2 \doteq `\solidline`, linha sólida
 - 3 \doteq `\dashline`, linha a tracejado
 - 4 \doteq `\dotline`, linha a ponteadado
 - 5 \doteq `\injectionarrow`, seta de injeção. Valor anterior 3 (versão < 4.2)
 - 6 \doteq `\applicationarrow`, seta de aplicação. Valor anterior 4 (versão < 4.2)
 - 7 \doteq `\surjectivearrow`, seta de função sobrejectiva. Valor anterior 5 (versão < 4.2)

`\mor(#1,#2)(#3,#4)[#5,#6]{#7}[#8,#9]`: Comando de colocação da seta (morfismo) de ligação de dois objectos – Segunda variante.

- #1 e #2 – coordenadas do nó de partida
- #3 e #4 – coordenadas do nó de chegada

Todos os outros argumentos têm o significado já explicado (por isso a numeração errada). É de notar que para a primeira variante é feito o cálculo das coordenadas dos nós de forma automática e depois são passados esses valores para a segunda variante do comando.

`\cmor(#1) #2(#3,#4){#5}[#6]` comando para a especificação de setas curvas. O algoritmo de construção das setas é o do `PiCTEX` o que implica que se está a especificar uma linha quadrática através de um número ímpar de pontos.

- #1—lista de pontos, em número ímpar
- #2—direccionamento da seta
 - 0 \doteq `\pup`, apontar para cima
 - 1 \doteq `\pdown`, apontar para baixo
 - 2 \doteq `\pright`, apontar para a direita
 - 3 \doteq `\pleft`, apontar para a esquerda
- #3—abscissa do morfismo
- #4—ordenada do morfismo
- #5—morfismo
- #6—tipo de “seta”, valor por omissão: 0, seta sólida.

Os restantes valores possíveis são os descritos na variante anterior.

O comando `cmor` no caso em que não tem o último parâmetro opcional tem de ser seguido por um espaço. O espaço antes do direccionamento da seta é obrigatório.

No caso de se ter o valor 2 (“`\solidline`”) o valor para o direccionamento da seta não é tipo em conta, no entanto dado se tratar de um do parâmetro obrigatório é necessário dar-lhe um valor

5 Desenvolvimento Futuro

Uma alteração de fundo será o de substituir o `PiCTEX` por outro “motor gráfico” eventualmente mais poderoso, por exemplo o `Metapost`, para já isso é só um pensamento distante.

6 Alguns Exemplos

6.1 Setas Duplas, Transformações Naturais, ...

É de notar que alguns casos aparentemente omissos na actual versão podem perfeitamente ser construídos através de uma utilização imaginativa dos actuais comandos. Por exemplo os seguintes diagramas:

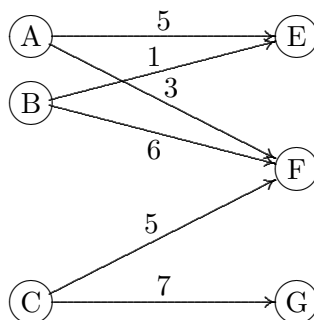
$$A \begin{array}{c} \xrightarrow{g} \\ \xrightarrow{f} \end{array} B \qquad A \begin{array}{c} \xrightarrow{\quad} \\ \downarrow \sigma \\ \xrightarrow{\quad} \\ \downarrow \tau \\ \xrightarrow{\quad} \end{array} B$$

Podem ser construídos com a actual versão. Eis como:

```
\begin{cd}[1]
\obj(5,5){A}
\obj(20,5){B}
\mor(5,4)(20,4){f}[\atright,\solidarrow]
\mor(5,6)(20,6){g}
\end{cd}

\begin{cd}[14]
\obj(5,5){A}
\obj(9,5){B}
\mor(5,6)(9,6){\downarrow\sigma}[\atright,\solidarrow]
\mor{A}{B}{}
\mor(5,4)(9,4){\downarrow\tau}
\end{cd}
```

6.2 Grafos Orientados com Objectos Circunscritos



Foi produzido através dos seguintes comandos:

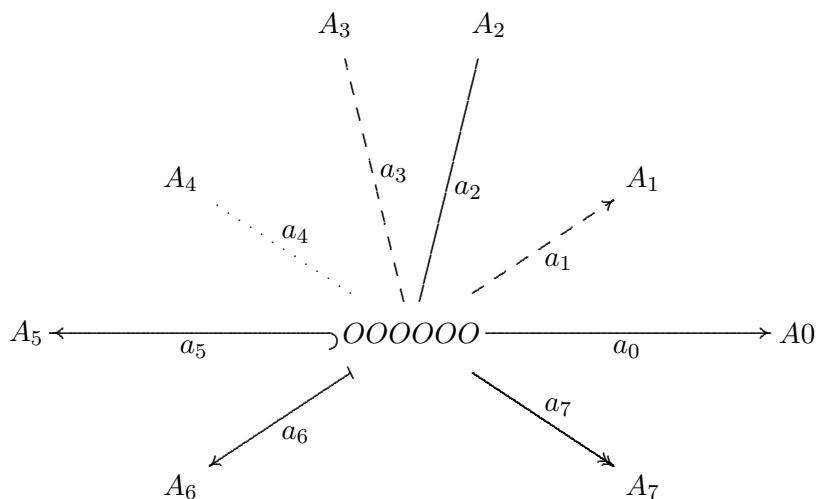
```
\begin{cd}[25]
\obj(1,5){A}
\obj(1,4){B}
\obj(1,1){C}
\obj(5,5){E}
\obj(5,3){F}
\obj(5,1){G}
\mor{A}{E}[8,8]{5}
\mor{A}{F}[8,8]{3}
\mor{B}{F}[8,8]{6}[\atright,\solidarrow]
\mor{B}{E}[8,8]{1}
\mor{C}{F}[8,8]{5}
\mor{C}{G}[8,8]{7}
\end{cd}
```

6.3 Diferentes Tipos de Setas/Linhas

```

\begin{cd}[29]
\obj(10,10)[A]{\$00000\$}
\obj(15,10)[A0]{\$A0\$}
\obj(13,12)[A1]{\$A_1\$}
\obj(11,14)[A2]{\$A_2\$}
\obj(9,14)[A3]{\$A_3\$}
\obj(7,12)[A4]{\$A_4\$}
\obj(5,10)[A5]{\$A_5\$}
\obj(7,8)[A6]{\$A_6\$}
\obj(13,8)[A7]{\$A_7\$}
\mor{A}{A0}{\$a_0\$}[\atright,\solidarrow]
\mor{A}{A1}{\$a_1\$}[\atright,\dashArrow]
\mor{A}{A2}{\$a_2\$}[\atright,\solidline]
\mor{A}{A3}{\$a_3\$}[\atright,\dashline]
\mor{A}{A4}{\$a_4\$}[\atright,\dotline]
\mor{A}{A5}{\$a_5\$}[\atleft,\injectionarrow]
\mor{A}{A6}{\$a_6\$}[\atleft,\aplicationarrow]
\mor{A}{A7}{\$a_7\$}[\atleft,\surjectivearrow]
\end{cd}

```



6.4 Diagramas com Setas Curvas

```

\begin{cd}[3]
\obj(14,11){\$A\$}
\obj(39,11){\$B\$}
\mor(14,12)(39,12){\$f\$}
\mor(39,10)(14,10){\$g\$}
\cmor((10,10)(6,11)(5,15)(6,19)(10,20)(14,19)(15,15))
\pdown(2,20){\$id_A\$}
\cmor((40,7)(41,3)(45,2)(49,3)(50,7)(49,11)(45,12))
\pleft(54,3){\$id_B\$}
\end{cd}

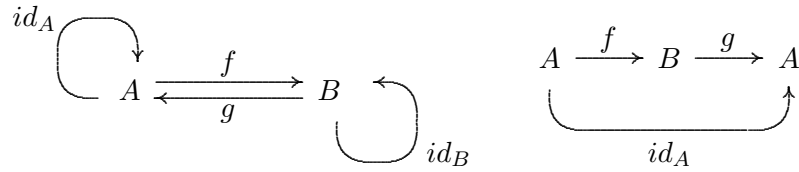
```

```

\begin{cd}[3]
\obj(10,15)[A]{\$A\$}
\obj(40,15)[Aa]{\$A\$}
\obj(25,15)[B]{\$B\$}
\mor{A}{B}{\$f\$}
\mor{B}{Aa}{\$g\$}
\cmor((10,11)(11,7)(15,6)(25,6)(35,6)(39,7)(40,11))
\pup(25,3){\$id_A\$}

```


\enddc



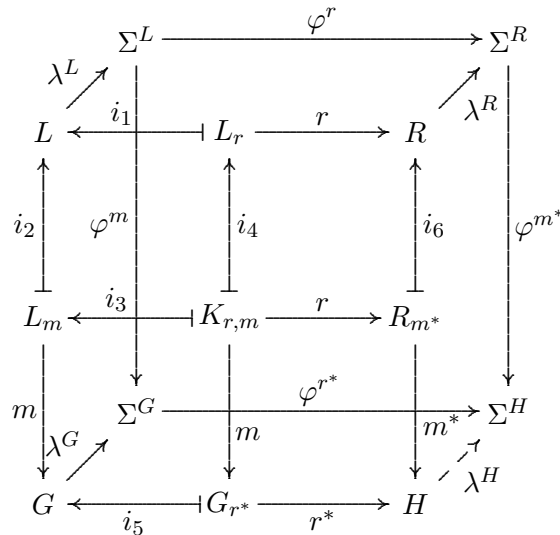
6.5 Um Exemplo Complexo

O diagrama seguinte foi proposto por Feruglio (Feruglio, 1994) como um caso de teste. Como é possível ver o DCpic produz o diagrama correctamente a partir de uma especificação simples.

```

\newcommand{\barraA}{\vrule height2em width0em depth0em}
\newcommand{\barraB}{\vrule height1.6em width0em depth0em}
\begindc{\commdiag}[35]
\obj(1,1)[Gr]{$G$}
\obj(3,1)[Grstar]{$G_{r^*}$}
\obj(5,1)[H]{$H$}
\obj(2,2)[SigmaG]{$\Sigma^G$}
\obj(6,2)[SigmaH]{$\Sigma^H$}
\obj(1,3)[Lm]{$L_m$}
\obj(3,3)[Krm]{$K_{r,m}$}
\obj(5,3)[Rmstar]{$R_{m^*}$}
\obj(1,5)[L]{$L$}
\obj(3,5)[Lr]{$L_r$}
\obj(5,5)[R]{$R$}
\obj(2,6)[SigmaL]{$\Sigma^L$}
\obj(6,6)[SigmaR]{$\Sigma^R$}
\mor{Gr}{SigmaG}{$\lambda^G$}
\mor{Grstar}{Gr}{$i_5$}[\atleft,\aplicationarrow]
\mor{Grstar}{H}{$r^*$}[\atright,\solidarrow]
\mor{H}{SigmaH}{$\lambda^H$}[\atright,\dashArrow]
\mor{SigmaG}{SigmaH}{$\varphi^{r^*}$}[\atleft,\solidarrow]
\mor{Lm}{Gr}{$m$}[\atright,\solidarrow]
\mor{Lm}{L}{$i_2$}[\atleft,\aplicationarrow]
\mor{Krm}{Lm}{$i_3\quad$}[\atright,\aplicationarrow]
\mor{Krm}{Rmstar}{$r$}
\mor{Krm}{Lr}{$i_4$}[\atright,\aplicationarrow]
\mor{Krm}{Grstar}{\barraA$m$}
\mor{Rmstar}{R}{$i_6$}[\atright,\aplicationarrow]
\mor{Rmstar}{H}{\barraB$m^*$}
\mor{L}{SigmaL}{$\lambda^L$}
\mor{Lr}{L}{$i_1\quad$}[\atright,\aplicationarrow]
\mor{Lr}{R}{$r$}
\mor{R}{SigmaR}{$\lambda^R$}[\atright,\solidarrow]
\mor{SigmaL}{SigmaG}{$\varphi^m$}[\atright,\solidarrow]
\mor{SigmaL}{SigmaR}{$\varphi^r$}
\mor{SigmaR}{SigmaH}{$\varphi^{m^*}$}
\enddc

```



Referências

- FERUGLIO, GABRIEL VALIENTE. 1994. Typesetting Commutative Diagrams. *TUGboat*, **15**(4), 466–484.
- HARARY, FRANK. 1972. *Graph Theory*. Reading, Massachusetts: Addison-Wesley.
- HERRLICH, HORST, & STRECKER, GEORGE. 1973. *Category Theory*. Allyn and Bacon Inc.
- KNUTH, DONALD E. 1986. *The T_EXbook*. Reading, Massachusetts: Addison-Wesley Publishing Company.
- LAMPORT, LESLIE. 1994. *L^AT_EX: A Document Preparation System*. 2nd edn. Reading, Massachusetts: Addison-Wesley Publishing Company.
- PIERCE, BENJAMIN. 1998. *Basic Category Theory for Computer Scientists*. Foundations of Computing. London, England: The MIT Press.

A O Código

```

%% DC-PiCTeX
%% Copyright (c) 1990-2009 Pedro Quaresma, Coimbra, Portugal
%% 11/1990 (version 1.0);
%% 10/1991 (version 1.1);
%% 9/1993 (version 1.2);
%% 3/1995 (version 1.3);
%% 7/1996 (version 2.1);
%% 5/2001 (version 3.0);
%% 11/2001 (version 3.1);
%% 1/2002 (version 3.2)
%% 5/2002 (version 4.0);
%% 3/2003 (version 4.1);
%% 12/2004 (version 4.1.1)
%% 3/2007 (version 4.2)
%% 5/2008 (version 4.2.1)
%% 8/2008 (version 4.3)
%% 12/2008 (version 4.3.1)
%% 12/2009 (version 4.3.2)

\immediate\write10{Package DCpic 2009/12/22 v4.3.2}

\ProvidesPackage{dcpic}[2009/12/22 v4.3.2]

%% Version X.Y.Z
%% X - major versions
%% Y - minor versions
%% Z - bug corrections
%%

```

```

%% Copyright (c) 1990-2009 Pedro Quaresma <pedro@mat.uc.pt>
%%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either version 1.3
% of this license or (at your option) any later version.
% The latest version of this license is in
% http://www.latex-project.org/lppl.txt
% and version 1.3 or later is part of all distributions of LaTeX
% version 2005/12/01 or later.
%
% This work has the LPPL maintenance status 'maintained'.
%
% The Current Maintainer of this work is Pedro Quaresma (pedro@mat.uc.pt).
%
% This work consists of the files dcpic.sty.
%%
%% Coimbra, 22 de Dezembro de 2009 (2009/12/22)
%% Pedro Quaresma de Almeida
%%
%% DCpic is a package of \TeX\ macros for graph modelling in a
%% (La)\TeX\ or ConTeXt document. Its distinguishing features are:
%% the use of \PiCTeX\ a powerful graphical engine, and a simple
%% specification syntax. A graph is described in terms of its objects
%% and its edges. The objects are textual elements and the edges can
%% have various straight or curved forms.
%%
%%
%% A graph in DCpic is a "picture" in \PiCTeX, in which we place our
%% {\em objects} and {\em morphisms} (edges). The user's commands in
%% DCpic are: {\tt begindc} and {\tt enddc} which establish the
%% coordinate system where the objects will be placed; {\tt obj}, the
%% command which defines the place and the contents of each object;
%% {\tt mor}, and {\tt cmor}, the commands which define the
%% morphisms, linear and curved edges, and its labels.
%%
%% Example:
%% \begin{commdiag}[3]
%% \obj(10,15){$A$}
%% \obj(25,15){$B$}
%% \obj(40,15){$C$}
%% \mor{$A$}{$B$}{$f$}
%% \mor{$B$}{$C$}{$g$}
%% \cmor((10,11)(11,7)(15,6)(25,6)(35,6)(39,7)(40,11))
%% \pup(25,3){$g\circ f$}
%% \enddc
%%
%% NOTES:
%% all the numeric values should be integer values.
%%
%% Available commands:
%%
%% The environment:
%% \begin{#1}[#2]
%% #1 - Graph type
%% 0 = "commdiag" (commutative diagram)
%% 1 = "digraph" (direct graph)
%% 2 = "undigraph" (undirect graph)
%% 3 = "cdigraph" with incircled objects
%% 4 = "cundigraph" with incircled objects
%% (optional) #2 - magnification factor (default value, 30)
%%
%% \enddc
%%
%% Objects:
%% \obj(#1,#2)[#3]{#4}[#5]
%% #1 and #2 - coordinates
%% (optional) #3 - Label, to be used in the morphisms command, if not
%% present the #4 will be used to that purpose
%% #4 - Object contents
%% (optional) #5 - placement of the object (default value \north)
%% 0="pcent", center
%% 1="north", north
%% 2="northeast", northeast
%% 3="east", east
%% 4="southeast", southeast

```

```

%%          5="\south", south
%%          6="\southwest", southwest
%%          7="\west", west
%%          8="\northwest", northwest
%% !!! Note !!!
%% if you omit the #3 argument (label) and the #4 argument is a
%% complex LaTeX command this can cause this command to crash. In
%% this case you must specify a label (the empty label [], if you do
%% needed it it for nothing).
%%
%% Morphims (linear edges). This command has to two major variants
%% i) Starting and Ending objects specification
%% \mor{#1}{#2}[#5,#6]{#7}[#8,#9]
%%
%% As you can see this first form is (intencionaly) badly formed, the
%% arguments #3 and #4 are missing (the actual command is correctly
%% formed).
%%
%%          #1 - The starting object reference
%%          #2 - The ending object reference
%%
%% from this two we will obtain the objects coordinates, and also the
%% dimensions of the enclosing box.
%%
%% The objects box dimensions are used to do an automatic adjustment of
%% the edge width.
%%
%% from #1 we obtain (x,y), (#1,#2) in the second form
%% from #2 we obtain (x',y'), (#3,#4) in the second form
%%
%% this values will be passed to the command second form
%%
%%ii) Two points coordinates specification
%% \mor{#1,#2}(#3,#4)[#5,#6]{#7}[#8,#9]
%%
%% Now we can describe all the arguments
%%
%%          #1 and #2 - coordinates (beginning)
%%          #3 and #4 - coordinates (ending)
%%(optional)#5,#6 - correction factors (default values, 10 and 10)
%%          #5 - actual beginning of the edge
%%          #6 - actual ending of the edge
%%          #7 - text (morphism label)
%%(optional)#8,#9
%%          #8 - label placement
%%              1 = "\atright", at right, default value
%%              -1 = "\atleft", at left
%%          #9 - edge type
%%              0 = "\solidarrow", default edge
%%              1 = "\dashArrow"
%%              2 = "\solidline"
%%              3 = "\dashline"
%%              4 = "\dotline"
%%              5 = "\injectionarrow" --- old value 3
%%              6 = "\applicationarrow" --- old value 4
%%              7 = "\surjectivearrow" --- old value 5
%%              8 = "\equalline" (thanks to Ruben Debeerst),
%%              9 = "\doublearrow"
%%              10 = "\doubleopposite"
%%              11 = "\nullarrow" (to allow adding labels to existing arrows)
%%
%% Notes: the equalline "arrow" does not provide a second label.
%%
%% Curved Morphisms (quadratic edges):
%% \cmor{#1} #2(#3,#4){#5}[#6]
%%          #1 - list of points (odd number)
%%          #2 - tip direction
%%              0 = "\pup", pointing up
%%              1 = "\pdown", pointing down
%%              2 = "\pright", pointing right
%%              3 = "\pleft", pointing left
%%          #3 and #4 - coordenates of the label
%%          #5 - morphism label
%%(optional)#6 - edge type
%%              0 = "\solidarrow", default value

```

```

%%          1 = "\dashArrow"
%%          2 = "\solidline"
%%
%% Notes: insert a space after the command.
%%        the space after the list of points is mandatory
%%
%% Exemples:
%%\documentclass[a4paper,11pt]{article}
%%\usepackage{dcpic,pictexwd}
%%
%%\begin{document}
%%\begin{dc}[3]
%%\obj(14,11){$A$}
%%\obj(39,11){$B$}
%%\mor(14,12)(39,12){$f$}[\atrightright,\solidarrow]
%%\mor(39,10)(14,10){$g$}[\atrightright,\solidarrow]
%%\cmor((10,10)(6,11)(5,15)(6,19)(10,20)(14,19)(15,15))
%% \pdown(2,20){$id_A$}
%%\cmor((40,7)(41,3)(45,2)(49,3)(50,7)(49,11)(45,12))
%% \pleft(54,3){$id_B$}
%%\end{dc}
%%
%%\begin{dc}{\commdiag}[3]
%%\obj(10,15)[A]{$A$}
%%\obj(40,15)[Aa]{$Aa$}
%%\obj(25,15)[B]{$B$}
%%\mor{A}{B}{$f$}[\atrightright,\solidarrow]
%%\mor{B}{Aa}{$g$}[\atrightright,\solidarrow]
%%\cmor((10,11)(11,7)(15,6)(25,6)(35,6)(39,7)(40,11))
%% \pup(25,3){$id_A$}
%%\end{dc}
%%
%%\newcommand{\barraA}{\vrule height2em width0em depth0em}
%%\newcommand{\barraB}{\vrule height1.6em width0em depth0em}
%%\begin{dc}{\commdiag}[35]
%%\obj(1,1)[Gr]{$G$}
%%\obj(3,1)[Grstar]{$G_{r^*}$}
%%\obj(5,1)[H]{$H$}
%%\obj(2,2)[SigmaG]{$\Sigma^G$}
%%\obj(6,2)[SigmaH]{$\Sigma^H$}
%%\obj(1,3)[Lm]{$L_m$}
%%\obj(3,3)[Krm]{$K_{r,m}$}
%%\obj(5,3)[Rmstar]{$R_{m^*}$}
%%\obj(1,5)[L]{$L$}
%%\obj(3,5)[Lr]{$L_r$}
%%\obj(5,5)[R]{$R$}
%%\obj(2,6)[SigmaL]{$\Sigma^L$}
%%\obj(6,6)[SigmaR]{$\Sigma^R$}
%%\mor{Gr}{SigmaG}{$\lambda^G$}
%%\mor{Grstar}{Gr}{$i_5$}[\atleftleft,\aplicationarrow]
%%\mor{Grstar}{H}{$r^*$}[\atrightright,\solidarrow]
%%\mor{H}{SigmaH}{$\lambda^H$}[\atrightright,\dashArrow]
%%\mor{SigmaG}{SigmaH}{$\varphi^{r^*}$}[\atrightright,\solidarrow]
%%\mor{Lm}{Gr}{$m$}[\atrightright,\solidarrow]
%%\mor{Lm}{L}{$i_2$}[\atleftleft,\aplicationarrow]
%%\mor{Krm}{Lm}{$i_3\quad$}[\atrightright,\aplicationarrow]
%%\mor{Krm}{Rmstar}{$r$}
%%\mor{Krm}{Lr}{$i_4$}[\atrightright,\aplicationarrow]
%%\mor{Krm}{Grstar}{$m$}
%%\mor{Rmstar}{R}{$i_6$}[\atrightright,\aplicationarrow]
%%\mor{Rmstar}{H}{$m^*$}
%%\mor{L}{SigmaL}{$\lambda^L$}
%%\mor{Lr}{L}{$i_1\quad$}[\atrightright,\aplicationarrow]
%%\mor{Lr}{R}{$r$}
%%\mor{R}{SigmaR}{$\lambda^R$}[\atrightright,\solidarrow]
%%\mor{SigmaL}{SigmaG}{$\varphi^m$}[\atrightright,\solidarrow]
%%\mor{SigmaL}{SigmaR}{$\varphi^r$}
%%\mor{SigmaR}{SigmaH}{$\varphi^{m^*}$}
%%\end{dc}
%%
%%\end{document}
%%-----//-----
%% Modifications (9/1993)
%%      argument "distance" between de tip of the arrow and the objects
%%      became optional; a new option for the "arrows" (option 3)

```

```

%%
%% 2/3/1995 (version 1.3)
%% adds "the aplicacion arrow" (option 4); the distance between
%% the label and the "arrow" is now a fixed value (10 units).
%% 15/7/1996 (version 2.1)
%% The comand "\mor" has a new syntax. The 5th and 6th
%% parameters are now the distance between the two objects and
%% the arrow tips. The 7th parameter is the label. The 8th e 9th
%% parameters (label position and type of arrow) are now optional
%%
%% 5/2001 (version 3.0)
%% Implementation of the comand "\cmor" based on the quadratic
%% curver comand of PiCTeX
%%
%% 11/2001 (version 3.1)
%% Changes on the tips of the arrow to became more LaTeX style
%% (after a conversation on EuroTeX 2001).
%%
%% 1/2002 (version 3.2)
%% Modifica{\c c}{\~a}o dos comandos "obj" e "mor" de forma a introducir
%% a especifica{\c c}{\~a}o l{\'o}gica dos morfismos, isto {\'e}, passa-se a
%% dizer qual {\'e} o objecto de partida e/ou o objecto de chegada
%% em vez de ter de especificar o morfismo em termos de
%% coordenadas.
%% Por outro lado o tamanho das setas passa a ser ajustado
%% automaticamente em rela{\c c}{\~a}o ao tamanho dos objectos.
%%
%% 5/2002 (version 4.0)
%% Modifica\~a\~a\~a dos comandos "begindc" e "obj"
%% !!! Modifica\~a\~a\~a da sintaxe destes comandos !!!
%% O comando "begindc" passou a ter um argumento (obrigat\~a\~a\~a)
%% que nos permite especificar o tipo de grafo que estamos a
%% querer especificar:
%% "commdiag" (0), para diagramas comutativos
%% "digraph" (1), para grafos orientados
%% "undigraph" (2), para grafos n\~a\~a\~a orientados
%% O comando "obj" modificou a sua sintaxe passou a ter um (ap\~a\~a\~a
%% a especifica\~a\~a\~a das coordenadas, um argumento opcional, um
%% argumento obrigat\~a\~a\~a, e um argumento opcional. O primeiro
%% argumento opcional d\~a\~a\~a-nos a etiqueta que serve como
%% refer\~a\~a\~a para a especifica\~a\~a\~a dos morfismos, na sua
%% aus\~a\~a\~a usa-se o argumento obrigat\~a\~a\~a para esse efeito, o
%% argumento obrigat\~a\~a\~a d\~a\~a\~a-nos o "conte\~a\~a\~a" do objecto, nos
%% diagramas comutativos \~a\~a\~a centrado no ponto dado pelas
%% coordenadas sendo o argumento seguinte simplesmente ignorado,
%% nos grafos o "conte\~a\~a\~a" \~a\~a\~a colocado numa posi\~a\~a\~a a norte, a
%% noroeste, a este, ..., sendo que a posi\~a\~a\~a concreta \~a\~a\~a
%% especificada pelo \~a\~a\~a\~a\~a\~a dos argumentos deste comando, o
%% valor por omis\~a\~a\~a \~a\~a\~a o "norte".
%%
%% 3/2003 (version 4.1)
%% Responding to a request of Jon Barker <jeb1@soton.ac.uk> I
%% create a new type of arrow, the surjective arrow.
%% For now only horizontal and vertical versions, other angles
%% are poorly rendered.
%% 12/2004 (version 4.1.1)
%% New version for the surjective arrows, solve the problems
%% with the first implementation of this option.
%% 3/2007 (version 4.2)
%% Adds the "providespackage" directive that was missing.
%% Adds dashed lines, and dotted lines.
%% 5/2008 (version 4.2.1)
%% Deleting some counters, trying to avoid the problem "running
%% out of counters", that occurs because of the use of PiCTeX
%% and DCPic (only two...)
%% 8/2008 (version 4.3)
%% Thanks to Ruben Debeerst (debeerst@mathematik.uni-kassel.de),
%% he added a new arrow "equalline". After that I
%% decided to add: the doublearrow; the doublearrow with
%% opposite directions; the null arrow. This last can be used as
%% a simple form of adding new labels.
%%12/2008 (version 4.3.1)
%% The comand \id is internalised (\!id), it should be that way
%% from the begining because it is not to be used from the
%% outside.

```

```

%%      The comand \dasharrow was changed to \dashArrow to avoid a
%%      clash with the AMS command with the same name.
%%12/2009 (version 4.3.2)
%%      There is a conflict between dcpic.sty and hyperref in current
%%      texlive-2009 due to the one letter macro \d (thanks Thorsten
%%      S <thorsten.schwander@gmail.com>).
%%      The \d changed to \deuc (Euclidian Distance). The \x and \y
%%      changed to \x0 \y0
%%-----//-----

```

```

\catcode'\!=11 % ***** THIS MUST NEVER BE OMITTED (See PiCTeX)

```

```

\newcount\aux%
\newcount\auxa%
\newcount\auxb%
\newcount\x0%
\newcount\y0%
\newcount\x1%
\newcount\y1%
\newcount\deuc%
\newcount\dnm%
\newcount\xa%
\newcount\xb%
\newcount\xmed%
\newcount\xc%
\newcount\xd%
\newcount\xe
\newcount\xf
\newcount\ya%
\newcount\yb%
\newcount\ymed%
\newcount\yc%
\newcount\yd
\newcount\ye
\newcount\yf
%% "global variables"
\newcount\expansao%
\newcount\tipografo%      version 4.0
\newcount\distanciaobjmor% version 4.0
\newcount\tipoarco%      version 4.0
%\newif\ifarredondada%    version 4.0 (initial value "false")
\newif\ifpara%
%% version 3.2
\newbox\caixa%
\newbox\caixaaux%
\newif\ifnvazia%
\newif\ifvazia%
\newif\ifcompara%
\newif\ifdiferentes%
\newcount\xaux%
\newcount\yaux%
\newcount\guardaauxa%
\newcount\alt%
\newcount\larg%
\newcount\prof%
%% for the trimming
\newcount\auxqx
\newcount\auxqy
\newif\ifajusta%
\newif\ifajustadist
\def\objPartida{}%
\def\objChegada{}%
\def\objNulo{}%

%%
%% Stack specification
%%
%%
%% Empty stack
%%
\def\!vazia{:}

%%

```

```

%% Is Empty? : Stack -> Bool
%%
%% nvazia - True if Not Empty
%% vazia - True if Empty
\def\!pilhanvazia#1{\let\arg=#1%
\if:\arg\!nvaziafalse\vaziatriue\else\nvaziatriue\vaziafalse\fi}

%%
%% Push : Elems x Stack -> Stack
%%
\def\!coloca#1#2{\edef\pilha{#1.#2}}

%%
%% Top : Stack -> Elems
%%
%% the empty stack is not taken care
%% the element is "kept" ("guardado")
\def\!guarda(#1)(#2,#3)(#4,#5,#6){\def\!id{#1}%
\iaux=#2%
\yaux=#3%
\alt=#4%
\larg=#5%
\prof=#6%
}

\def\!topaux#1.#2:{\!guarda#1}
\def\!topo#1{\expandafter\!topaux#1}

%%
%% Pop : Stack -> Stack
%%
%% the empty stack is not taken care
\def\!popaux#1.#2:{\def\pilha{#2:}}
\def\!retira#1{\expandafter\!popaux#1}

%%
%% Compares words : Word x Word -> Bool
%%
%% compara - True if equal
%% diferentes - True if not equal
\def\!comparaaux#1#2{\let\argA=#1\let\argB=#2%
\ifx\argA\argB\comparatriue\diferentesfalse\else\comparafalse\diferentestriue\fi}

\def\!compara#1#2{\!comparaaux{#1}{#2}}

%% Private Macro
%% Absolute Value)
%% \absoluto{n}{absn}
%% input
%% n - integer
%% output
%% absn - |n|
\def\!absoluto#1#2{\iaux=#1%
\ifnum\iaux > 0
#2=\iaux
\else
\multiply\iaux by -1
#2=\iaux
\fi}

%% Name definitions for edge types and directions
\def\solidarrow{0}
\def\dashArrow{1}
\def\solidline{2}
\def\dashline{3}
\def\dotline{4}
\def\injectionarrow{5}
\def\applicationarrow{6}
\def\surjectivearrow{7}
\def\equalline{8}
\def\doublearrow{9}
\def\doubleopposite{10}
\def>nullarrow{11}

```



```

%% Name definitions for edge label placement
\def\atright{-1}
\def\atleft{1}
%% Tip direction for curved edges
\def\pup{0}
\def\pdown{1}
\def\pright{2}
\def\pleft{3}
%% Type of graph
\def\commdiag{0}
\def\digraph{1}
\def\undigraph{2}
\def\cdigraph{3}
\def\cundigraph{4}
%% Positioning of labels in graphs
\def\pcent{0}
\def\north{1}
\def\northeast{2}
\def\east{3}
\def\southeast{4}
\def\south{5}
\def\southwest{6}
\def\west{7}
\def\northwest{8}

%%Comando Interno
%% Ajusta a dist{\^a}ncia entre as setas e os objectos em fun{\c c}{\^a}o das
%% dimens{\^o}es destes {\^u}ltimos
%% \ajusta{x}{xl}{y}{yl}{d}{Objecto}
%% entrada
%% (x,y) e (xl,yl), coordenadas dos pontos de {\^i}ncio e fim da seta
%% d, dist{\^a}ncia especificada pelo utilizador ou 10 (valor por
%% omiss{\^a}o), Objecto d{\^a}-nos a refer{\^e}ncia do objecto ao qual se est{\^a} a
%% efectuar o ajuste.
%% sa{\^i}da
%% d, dist{\^a}ncia alterada.
%%
%% A dist{\^a}ncia alterada {\^e} o maior valor entre 10 e as dimens{\^o}es
%% apropriadas da caixa que cont{\^e}m o objecto.
%% Se o utilizador especificar um valor essa especifica{\c c}{\^a}o
%% n{\^a}o {\^e} alterada.
%%
%% Se a seta {\^e} horizontal usa-se o valor da largura
%% Se a seta {\^e} vertical usa-se:
%% o valor da altura se a seta est{\^a} no 1o ou 2o quadrante
%% o valor da profundidade se a seta est{\^a} no 3o ou 4o quadrante
%% Se a seta {\^e} {\^o}bliqua vai-se escolher o valor conforme:
%% de 315 a 45 graus usa-se a largura
%% de 45 a 135 graus usa-se a altura
%% de 135 a 225 graus usa-se a largura
%% de 225 a 315 graus usa-se a profundidade
\def\ajusta#1#2#3#4#5#6{\aux=#5%
\let\auxobj=#6%
\ifcase \tipografo % diagramas comutativos
\ifnum\number\aux=10
\ajustadisttrue % se o valor \^A1' o valor por omiss\^Aço ajusta
\else
\ajustadistfalse % caso contr\^Ario n\^Aço ajusta
\fi
\else % grafos (dirigidos, n\^Aço dirigidos, com molduras)
\ajustadistfalse
% \or % grafos n\^Aço dirigidos
% \ajustadistfalse
% \else % grafos dirigidos com molduras circulares nos objectos
% \ifnum\number\aux=8
% \ajustadisttrue % se o valor \^A1' o valor por omiss\^Aço ajusta
% \else
% \ajustadistfalse % caso contr\^Ario n\^Aço ajusta
% \fi
\fi
\ifajustadist
% \tiny Vou ajustar %%%
% \ifnum\number\aux=10% verificar se s{\^a}o os valores por omiss{\^a}o

```

```

\let\pilhaaux=\pilha%
\loop%
  \!topo{\pilha}%
  \!retira{\pilha}%
  \!compara{\!id}{\auxobj}%
  \ifcompara\nvaziafalse \else\!pilhanvazia\pilha \fi%
  \ifnvazia%
\repeat%
%% rep{\~o}e os valores na pilha
\let\pilha=\pilhaaux%
\ifvazia%
  \ifdiferentes%
%%
%% N{\~a}o {\~e} poss{\~i}vel efectuar o ajuste dado o utilizador n{\~a}o ter
%% especificado uma etiqueta para o objecto em quest{\~a}o. {\~E} dado o
%% valor de 10, igual ao valor por omiss{\~a}o.
%%
  \larg=1310720% n{\~a}o faz o ajuste
  \prof=655360%
  \alt=655360%
  \fi%
\fi%
\divide\larg by 131072
\divide\prof by 65536
\divide\alt by 65536
\ifnum\number\y0=\number\y1
%% Caso 1 -- seta horizontal
%%
%% divide-se por 131072 para se obter metade da largura da caixa em
%% pontos (pt), isto dado que o texto est{\~a} centrado na caixa. Soma-se
%% mais tr{\~e}s, que constitue um ajuste imp{\~i}rico.
  \advance\larg by 3
  \ifnum\number\larg>\aux
    #5=\larg
  \fi
\else
  \ifnum\number\x0=\number\x1
  \ifnum\number\y1>\number\y0
%% Caso 2.1 -- seta vertical de cima para baixa
%%
  \ifnum\number\alt>\aux
    #5=\alt
  \fi
\else
%% Caso 2.2 -- seta vertical de baixo para cima
%%
%% divide-se por 65536 para se obter a altura da caixa em pt. 0 ajuste
%% de 5 foi obtido imp{\~i}ricamente
  \advance\prof by 5
  \ifnum\number\prof>\aux
    #5=\prof
  \fi
\fi
\else
%% Caso 3 -- seta obl{\~i}qua
%% Caso 3.1 de 315o a 45o; |x-x1|>|y-y1| e
%% Caso 3.3 de 135o a 225o; |x-x1|>|y-y1|; Largura
  \auxqx=\x0
  \advance\auxqx by -\x1
  \!absoluto{\auxqx}{\auxqx}%
  \auxqy=\y0
  \advance\auxqy by -\y1
  \!absoluto{\auxqy}{\auxqy}%
  \ifnum\auxqx>\auxqy
  \ifnum\larg<10
    \larg=10
  \fi
  \advance\larg by 3
  #5=\larg
\else
%% Caso 3.2 de 45o a 135o; |x-x1|<|y-y1| e y>0; Largura
  \ifnum\y1>\y0
  \ifnum\larg<10
    \larg=10
  \fi

```

```

\advance\alt by 6
#5=\alt
\else
%% Caso 3.4 de 225o a 315o; |x-x1|<|y-y1| e y<0; Profundidade
\advance\prof by 11
#5=\prof
\fi
\fi
\fi
\fi
\fi} % o ramo "else" {\e} omisso

%%Comando Interno
%% C{\a}lculo da Raiz Quadrada
%% raiz{n}{m}
%% entrada
%% n - natural
%% sa{\i}da
%% n - natural
%% m - maior natural contido na raiz quadrada de n
\def\!raiz#1#2{\auxa=#1%
\auxb=1%
\loop
\aux=\auxb%
\advance \aux by 1%
\multiply \aux by \aux%
\ifnum \aux < \auxa%
\advance \auxb by 1%
\paratrue%
\else\ifnum \aux=\auxa%
\advance \auxb by 1%
\paratrue%
\else\parafalse%
\fi
\fi
\ifpara%
\repeat
#2=\auxb}

%%Private Macro
%% Find the starting and ending points of the "arrow" and also the
%% label position (one coordinate at a time)
%%
%% ucoord{x1}{x2}{x3}{x4}{x5}{x6}{+|- 1}
%% Input
%% x1,x2,x3,x4,x5
%% Output
%% x6
%%
%% x2 - x1
%% x6 = x3 +|- ----- x4
%% x5
\def\!ucoord#1#2#3#4#5#6#7{\aux=#2%
\advance \aux by -#1%
\multiply \aux by #4%
\divide \aux by #5%
\ifnum #7 = -1 \multiply \aux by -1 \fi%
\advance \aux by #3%
#6=\aux}

%%Private Macro
%% Euclidean distance between two points
%%
%% quadrado{n}{m}{l}
%% Input
%% n - natural number
%% m - natural number
%% Output
%% l = (n-m)*(n-m)
\def\!quadrado#1#2#3{\aux=#1%
\advance \aux by -#2%
\multiply \aux by \aux%
#3=\aux}

```

```

%%Comando Interno
%% C{\a}lculo auxiliar para determinar a dist{\a}ncia entre o nome do
%% morfismo e a seta.
%% entrada
%% (x,y), (x',y') e o nome do morfismo
%% sa{\i}da
%% dnm - dist{\a}ncia do nome ao morfismo respectivo devidamente
%% compensada pelo tamanho do objecto
%% Observa{\c c}{\o}es
%% A compensa{\c c}{\a}o s{\o} est{\a} a ser feita para setas
%% horizontais e verticais. As obl{\i}quas s{\a}o tratadas de
%% outra forma.
%% algoritmo
%% caixa0 <- nome do morfismo
%% se x-x1 = 0 entao {recta vertical}
%% aux <- largura da caixa0
%% dnm <- convers{\a}o-sp-pt(aux)/2+3
%% sen{\a}o {recta n{\a}o vertical}
%% se y-y1 = 0 entao {recta horizontal}
%% aux <- altura+profundidade da caixa0
%% dnm <- convers{\a}o-sp-pt(aux)/2+3
%% sen{\a}o {recta obl{\i}qua}
%% dnm <- 3
%% fimse
%% fimse
%% fimalgoritmo
\def\!distnomemor#1#2#3#4#5#6{\setbox0=\hbox{#5}%
\aux=#1
\advance \aux by -#3
\ifnum \aux=0
\aux=\wd0 \divide \aux by 131072
\advance \aux by 3
#6=\aux
\else
\aux=#2
\advance \aux by -#4
\ifnum \aux=0
\aux=\ht0 \advance \aux by \dp0 \divide \aux by 131072
\advance \aux by 3
#6=\aux%
\else
#6=3
\fi
\fi
}

%%
%% 0 ambiente "begindc...enddc"
%%
\def\begindc#1{\!ifnextchar[{\!begindc{#1}}{\!begindc{#1}[30]}}
\def\!begindc#1[#2]{\beginpicture
\let\pilha=\!vazia
\setcoordinatesystem units <1pt,1pt>
\expansao=#2
\ifcase #1
\distanciaobjmor=10
\tipoarco=0 % arrow
\tipografo=0 % commutative diagram
\or
\distanciaobjmor=2
\tipoarco=0 % arrow
\tipografo=1 % directed graph
\or
\distanciaobjmor=1
\tipoarco=2 % line
\tipografo=2 % undirected graph
\or
\distanciaobjmor=8
\tipoarco=0 % arrow
\tipografo=3 % directed graph
% \arredondadotru e % objectos com molduras circulares
\or
\distanciaobjmor=8
\tipoarco=2 % line

```

```

\tipografo=4      % undirected graph
% \arredondadotru  % objetos com molduras circulares
\fi}

\def\enddc{\endpicture}

%% Public macro: "mor"
%%
%%
%% Funtion to built the "arrow" between two points
%%
%% The points that are uses to built all the elements of the "arrows"
%% are:
%%
%%          (xc,yc)
%%          o
%%          |
%% o-----o-----o-----o-----o
%%(x,y) (xa,ya)  (xm,ym)  (xb,yb)(xl,y1)
%%
%% auxa - distance between (x,y) and (xa,ya), 10pt by default
%% auxb - distance between (xl,y1) and (xb,yb), 10pt by default
%%
\def\mor{%
  \!ifnextchar({\!morxy}{\!morObjA}}
\def\!morxy(#1,#2){%
  \!ifnextchar({\!morxyl{#1}{#2}}{\!morObjB{#1}{#2}})
\def\!morxyl#1#2(#3,#4){%
  \!ifnextchar[{\!mora{#1}{#2}{#3}{#4}}{\!mora{#1}{#2}{#3}{#4}[\number\distanciaobjmor,\number\distanciaobjmor]}]}%
\def\!morObjA#1{%
  \let\pilhaaux=\pilha%
  \def\objPartida{#1}%
  \loop%
    \!topo\pilha%
    \!retira\pilha%
    \!compara{\!id}{\objPartida}%
    \ifcompara \nvaziafalse \else \!pilhanvazia\pilha \fi%
    \ifnvazia%
  \repeat%
  \ifvazia%
  \endifdiferentes%
%%
%% error message and ficticious parameters
%%
Error: Incorrect label specification%
\aux=1%
\yaux=1%
\fi%
\fi%
\let\pilha=\pilhaaux%
\!ifnextchar({\!morxyl{\number\aux}{\number\yaux}}{\!morObjB{\number\aux}{\number\yaux}})
\def\!morObjB#1#2#3{%
  \x0=#1
  \y0=#2
  \def\objChegada{#3}%
  \let\pilhaaux=\pilha%
  \loop
    \!topo\pilha %
    \!retira\pilha%
    \!compara{\!id}{\objChegada}%
    \ifcompara \nvaziafalse \else \!pilhanvazia\pilha \fi
    \ifnvazia
  \repeat
  \ifvazia
  \endifdiferentes%
%%
%% error message and ficticious parameters
%%
Error: Incorrect label specification
\aux=\x0%
\advance\aux by \x0%
\yaux=\y0%
\advance\yaux by \y0%
\fi

```

```

\fi
\let\pilha=\pilhaaux
\!ifnextchar[{\!mora{\number\x0}{\number\y0}{\number\aux}{\number\yaux}}{\!mora{\number\x0}{\number\y0}{\number\aux}{\number\yaux}}
\def\!mora#1#2#3#4[#5,#6]#7{
\!ifnextchar[{\!morb{#1}{#2}{#3}{#4}{#5}{#6}{#7}}{\!morb{#1}{#2}{#3}{#4}{#5}{#6}{#7}[1,\number\tipoarco ]}
\def\!morb#1#2#3#4#5#6#7[#8,#9]{\x0=#1%
\y0=#2%
\x1=#3%
\y1=#4%
\multiply\x0 by \expansao%
\multiply\y0 by \expansao%
\multiply\x1 by \expansao%
\multiply\y1 by \expansao%
%%
%% Euclidean distance between two points
%% d = \sqrt((x-x1)^2+(y-y1)^2)
%%
\!quadrado{\number\x0}{\number\x1}{\auxa}%
\!quadrado{\number\y0}{\number\y1}{\auxb}%
\deuc=\auxa%
\advance\deuc by \auxb%
\!raiz{\deuc}{\deuc}%
%%
%% the point (xa,ya) is at a distance #5 (default value 10) from the
%% point (x,y)
%%
%% como existem dois pontos em considera{\c c}{\~a}o, o ponto de partida e o
%% ponto de chegada, vai sei necess{\'a}rio recuperar de novo os seus
%% valores por pesquisa na pilha
\auxa=#5
\!compara{\objNulo}{\objPartida}%
\ifdiferentes% S{\'o} vai fazer o ajuste quando {\'e} necess{\'a}rio
\!ajusta{\x0}{\x1}{\y0}{\y1}{\auxa}{\objPartida}%
\ajustatru
\def\objPartida{}% re-inicializar o valor do Objecto de Partida
\fi
%% vai guardar o valor de auxa (ap{\'o}s ajuste) para ser usado no caso
%% dos morfismos de injec{\c c}{\~a}o.
\guardaauxa=\auxa
%%
\!ucoord{\number\x0}{\number\x1}{\number\x0}{\auxa}{\number\deuc}{\xa}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\y0}{\auxa}{\number\deuc}{\ya}{1}%
%% auxa vai ter o valor da dist{\^a}ncia entre os objectos menos a
%% dist{\^a}ncia da seta ao objecto (10 por omiss{\~a}o)
\auxa=\deuc%
%%
%% the point (xb,yb) is at a distance #6 (default value 10) from the
%% point (x1,y1)
%%
\auxb=#6
\!compara{\objNulo}{\objChegada}%
\ifdiferentes% S{\'o} vai fazer o ajuste quando {\'e} necess{\'a}rio
% Vou ajustar
\!ajusta{\x0}{\x1}{\y0}{\y1}{\auxb}{\objChegada}%
\def\objChegada{}% re-inicializar o valor do Objecto de Chegada
\fi
\advance\auxa by -\auxb%
\!ucoord{\number\x0}{\number\x1}{\number\x0}{\number\auxa}{\number\deuc}{\xb}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\y0}{\number\auxa}{\number\deuc}{\yb}{1}%
\xmed=\xa%
\advance\xmed by \xb%
\divide\xmed by 2
\ymed=\ya%
\advance\ymed by \yb%
\divide\ymed by 2
%%
%% find the coordinates of the label position: (xc,yc)
%%
%% after this the values of xmed and ymed are no longer important
%%
\!distnomemor{\number\x0}{\number\y0}{\number\x1}{\number\y1}{#7}{\dnm}%
\!ucoord{\number\y0}{\number\y1}{\number\xmed}{\number\dnm}{\number\deuc}{\xc}{-#8}%
\!ucoord{\number\x0}{\number\x1}{\number\ymed}{\number\dnm}{\number\deuc}{\yc}{#8}%
%%
%% draw the "arrow"

```

```

%%
\ifcase #9           % 0=solid arrow
  \arrow <4pt> [.2,1.1] from {\xa} {\ya} to {\xb} {\yb}
\or
  % 1=dashed arrow
  \setdashes
  \arrow <4pt> [.2,1.1] from {\xa} {\ya} to {\xb} {\yb}
  \setsolid
\or
  % 2=solid line
  \setlinear
  \plot {\xa} {\ya} {\xb} {\yb} /
\or
  % 3=dashed line
  \setdashes
  \setlinear
  \plot {\xa} {\ya} {\xb} {\yb} /
  \setsolid
\or
  % 4=dotted line
  \setdots
  \setlinear
  \plot {\xa} {\ya} {\xb} {\yb} /
  \setsolid
\or
  % 5=injective arrow
%% C{'a}lculos auxiliares
%%
%% 3 units, the radius for the tail of the arrow
%%
%% repor o valor de auxa
  \auxa=\guardaauxa
%% dar a compensa{c c}{\~a}o para o "rabo"
  \advance \auxa by 3%
%%
%% Note: the values of (xa,ya) will be modified, they will be
%% "pushed" further away from (x,y) in order to acomodate the tail
%% of the "arrow"
%%
%% find the point (xd,yd), the center of a 3pt circle
%%
\!ucoord{\number\x0}{\number\x1}{\number\x0}{\number\auxa}{\number\deuc}{\xa}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\y0}{\number\auxa}{\number\deuc}{\ya}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xa}{3}{\number\deuc}{\xd}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{3}{\number\deuc}{\yd}{1}%
%% building the "arrow"
  \arrow <4pt> [.2,1.1] from {\xa} {\ya} to {\xb} {\yb}
%% and its "tail"
  \circulararc -180 degrees from {\xa} {\ya} center at {\xd} {\yd}
\or
  % 6=maps "arrow" ("|-->")
  \auxa=3
%%
%% Note: the values of xmed and ymed will be modified
%%
%% find the two points that defines the tail of the arrow (segment
%% (xmed,ymed)(xd,yd))
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xmed}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\ymed}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xd}{1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\yd}{-1}%
%% building the "arrow"
  \arrow <4pt> [.2,1.1] from {\xa} {\ya} to {\xb} {\yb}
%% and its "tail"
  \setlinear
  \plot {\xmed} {\ymed} {\xd} {\yd} /
\or
  % 7=surjective arrow ("-->>")
%% building arrow with the first tip
  \arrow <4pt> [.2,1.1] from {\xa} {\ya} to {\xb} {\yb}
%% and the second tip
  \setlinear
  \arrow <6pt> [0,.72] from {\xa} {\ya} to {\xb} {\yb}
\or
  % 8=equalline
%% by Ruben Debeerst: equal-line
%%
%% sets the separation (distance) between the two parallel lines, if
%% horizontal or vertical 1pt is enough, if not 2pt
\auxa=2
\ifnum\number\y0=\number\y1
\auxa=1
\fi

```

```

\ifnum\number\x0=\number\x1
\auxa=1
\fi
%% the two parallel lines will be given by (xmed,ymed)(xd,yd), and
%% (xe,ye)(xf,yf)
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xmed}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\ymed}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xd}{1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\yd}{-1}%

\!ucoord{\number\y0}{\number\y1}{\number\xb}{\number\auxa}{\number\deuc}{\xe}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\yb}{\number\auxa}{\number\deuc}{\ye}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xb}{\number\auxa}{\number\deuc}{\xf}{1}%
\!ucoord{\number\x0}{\number\x1}{\number\yb}{\number\auxa}{\number\deuc}{\yf}{-1}%
\setlinear
\plot {\xmed} {\ymed} {\xe} {\ye} /
\plot {\xd} {\yd} {\xf} {\yf} /
\or
% 9=double arrow
%%
%% sets the separation (distance) between the two parallel lines, if
%% horizontal or vertical 2pt is enough, if not 3pt. The extra space
%% is needed because of the arrow tip.
\auxa=3
\ifnum\number\y0=\number\y1
\auxa=2
\fi
\ifnum\number\x0=\number\x1
\auxa=2
\fi
%% the two parallel lines will be given by (xmed,ymed)(xd,yd), and
%% (xe,ye)(xf,yf)
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xmed}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\ymed}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xd}{1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\yd}{-1}%

\!ucoord{\number\y0}{\number\y1}{\number\xb}{\number\auxa}{\number\deuc}{\xe}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\yb}{\number\auxa}{\number\deuc}{\ye}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xb}{\number\auxa}{\number\deuc}{\xf}{1}%
\!ucoord{\number\x0}{\number\x1}{\number\yb}{\number\auxa}{\number\deuc}{\yf}{-1}%
\arrow <4pt> [.2,1.1] from {\xmed} {\ymed} to {\xe} {\ye}
\arrow <4pt> [.2,1.1] from {\xd} {\yd} to {\xf} {\yf}
\or
% 10=double arrow, opposite directions
%%
%% sets the separation (distance) between the two parallel lines, if
%% horizontal or vertical 2pt is enough, if not 3pt. The extra space
%% is needed because of the arrow tip.
\auxa=3
\ifnum\number\y0=\number\y1
\auxa=2
\fi
\ifnum\number\x0=\number\x1
\auxa=2
\fi
%% the two parallel lines will be given by (xmed,ymed)(xd,yd), and
%% (xe,ye)(xf,yf)
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xmed}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\ymed}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xa}{\number\auxa}{\number\deuc}{\xd}{1}%
\!ucoord{\number\x0}{\number\x1}{\number\ya}{\number\auxa}{\number\deuc}{\yd}{-1}%

\!ucoord{\number\y0}{\number\y1}{\number\xb}{\number\auxa}{\number\deuc}{\xe}{-1}%
\!ucoord{\number\x0}{\number\x1}{\number\yb}{\number\auxa}{\number\deuc}{\ye}{1}%
\!ucoord{\number\y0}{\number\y1}{\number\xb}{\number\auxa}{\number\deuc}{\xf}{1}%
\!ucoord{\number\x0}{\number\x1}{\number\yb}{\number\auxa}{\number\deuc}{\yf}{-1}%
\arrow <4pt> [.2,1.1] from {\xmed} {\ymed} to {\xe} {\ye}
\arrow <4pt> [.2,1.1] from {\xf} {\yf} to {\xd} {\yd}
\or
% 11=null arrow (no arrow, only a label)
%%
%% does not draw the arrow, it allows to put two labels in one "arrow"
%%
\fi
%% The label positioning.
%% Se os morfismos s{\~a}o horizontais ou verticais constro{\'\i} uma caixa
%% centrada no ponto pr{\'e}viamente calculado. Se as setas s{\~a}o

```



```

%% obl{\',i}guas coloca a caixa de forma a n{\^a}o colidir com o morfismo
%% tendo em aten{\c c}{\^a}o o quadrante assim como a posi{\c c}{\^a}o
%% relativa do morfismo e do respectivo nome.
\auxa=\x1
\advance \auxa by -\x0%
\ifnum \auxa=0
  \put {#7} at {\xc} {\yc}
\else
  \auxb=\y1
  \advance \auxb by -\y0%
  \ifnum \auxb=0 \put {#7} at {\xc} {\yc}
  \else
    \ifnum \auxa > 0
      \ifnum \auxb > 0
        \ifnum #8=1
          \put {#7} [rb] at {\xc} {\yc}
        \else
          \put {#7} [lt] at {\xc} {\yc}
        \fi
      \else
        \ifnum #8=1
          \put {#7} [lb] at {\xc} {\yc}
        \else
          \put {#7} [rt] at {\xc} {\yc}
        \fi
      \fi
    \else
      \ifnum \auxb > 0
        \ifnum #8=1
          \put {#7} [rt] at {\xc} {\yc}
        \else
          \put {#7} [lb] at {\xc} {\yc}
        \fi
      \else
        \ifnum #8=1
          \put {#7} [lt] at {\xc} {\yc}
        \else
          \put {#7} [rb] at {\xc} {\yc}
        \fi
      \fi
    \fi
  \fi
\fi
}

%%
%% Comando para construir a "seta" curvilinea entre dois objectos
%%
%% \cmor(<lista de pontos (n. impar)>){<etiqueta>}
%%
%% Em primeiro lugar {\'e} necess{\'a}rio modificar o comando plot de forma a
%% que a sintaxe de utiliza{\c c}{\^a}o do novo comando seja coerente com a
%% sintaxe dos restantes comandos
%%
\def\modifplot(#1{\!modifqcurve #1}
\def\!modifqcurve(#1,#2){\x0=#1%
  \y0=#2%
  \multiply \x0 by \expansao%
  \multiply \y0 by \expansao%
  \!start (\x0,\y0)
  \!modifQjoin}
\def\!modifQjoin(#1,#2)(#3,#4){\x0=#1%
  \y0=#2%
  \x1=#3%
  \y1=#4%
  \multiply \x0 by \expansao%
  \multiply \y0 by \expansao%
  \multiply \x1 by \expansao%
  \multiply \y1 by \expansao%
  \!qjoin (\x0,\y0) (\x1,\y1) % \!qjoin is defined in QUADRATIC
  \!ifnextchar{\!fim}{\!modifQjoin}}
\def\!fim){\ignorespaces}

%%
%% O comando para desenhar a seta vai receber a lista de pontos da qual

```

```

%% retira o {\u}ltimo par de pontos, dependente da escolha dada pelo
%% utilizador a seta vai ser desenhada para cima, para baixo, para a
%% direita ou para a esquerda
%%
\def\setaxy(#1{\!pontosxy #1}
\def\!pontosxy(#1,#2){%
  \!maispontosxy}
\def\!maispontosxy(#1,#2)(#3,#4){%
  \!ifnextchar{\!fimxy#3,#4}{\!maispontosxy}}
\def\!fimxy#1,#2){\x0=#1%
  \y0=#2
  \multiply \x0 by \expansao
  \multiply \y0 by \expansao
  \x1=\x0%
  \y1=\y0%
  \aux=1%
  \multiply \aux by \auxa%
  \advance\x1 by \aux%
  \aux=1%
  \multiply \aux by \auxb%
  \advance\y1 by \aux%
  \arrow <4pt> [.2,1.1] from {\x0} {\y0} to {\x1} {\y1}}

%%
%% Temos agora a defini{\c c}{\^a}o do comando "cmor"
%%
\def\cmor#1 #2(#3,#4)#5{%
  \!ifnextchar[{\!cmora{#1}{#2}{#3}{#4}{#5}}{\!cmora{#1}{#2}{#3}{#4}{#5}[0] ]}
\def\!cmora#1#2#3#4#5[#6]{%
  \ifcase #2% para cima "\pup" (pointing up)
    \auxa=0% x mant{\^e}m-se
    \auxb=1% o y "sobe"
  \or% para baixo "\pdown" (pointing down)
    \auxa=0% x mant{\^e}m-se
    \auxb=-1% o y "desce"
  \or% para a direita "\pright" (pointing right)
    \auxa=1% o x move-se para a direita
    \auxb=0% o y mant{\^e}m-se
  \or% para a esquerda "\pleft" (pointing left)
    \auxa=-1% o x move-se para a esquerda
    \auxb=0% o y mant{\^e}m-se
  \fi % constru{\c c}{\^a}o do arco
\ifcase #6 % arco (com seta) s{\^o}lido
  \modifplot#1% Desenhar o arco
  % constru{\c c}{\^a}o da seta
  \setaxy#1
\or % arco (com seta) a tracejado
  \setdashes
  \modifplot#1% Desenhar o arco
  \setaxy#1
  \setsolid
\or % arco sem seta
  \modifplot#1% Desenhar o arco
\fi % seta de injec{\c c}{\^a}o
%% coloca{\c c}{\^a}o da etiqueta do morfismo
\x0=#3%
\y0=#4%
\multiply \x0 by \expansao%
\multiply \y0 by \expansao%
\put {#5} at {\x0} {\y0}}

%%
%% Comando para construir os Objectos
%% \obj(x,y){<texto>}[<etiqueta>]
%%
\def\obj(#1,#2){%
  \!ifnextchar[{\!obja{#1}{#2}}{\!obja{#1}{#2}[Nulo]}]
\def\!obja#1#2[#3]#4{%
  \!ifnextchar[{\!objb{#1}{#2}{#3}{#4}}{\!objb{#1}{#2}{#3}{#4}[1]}]
\def\!objb#1#2#3#4[#5]{%
  \x0=#1%
  \y0=#2%
  \def\!pinta{\normalsize$\bullet$}% para definir o tamanho normal das pintas
  \def\!nulo{Nulo}%

```

```

\def\!arg{#3}%
\!compara{\!arg}{\!nulo}%
\ifcompara\def\!arg{#4}\fi%
\multiply \x0 by \expansao%
\multiply \y0 by \expansao%
\setbox\caixa=\hbox{#4}%
\!coloca(\!arg)(#1,#2)(\number\ht\caixa,\number\wd\caixa,\number\dp\caixa)}{\pilha}%
\auxa=\wd\caixa \divide \auxa by 131072
\advance \auxa by 5
\auxb=\ht\caixa
\advance \auxb by \number\dp\caixa
\divide \auxb by 131072
\advance \auxb by 5
%(\number\auxa,
%\number\auxb)
% \aux=\ht\caixa \divide \auxa by 131072
% \advance \auxa by 5
% \auxb=\dp\caixa \divide \auxb by 131072
% \advance \auxb by 8
\ifcase \tipografo % diagramas comutativos
\put{#4} at {\x0} {\y0}
\or % grafos dirigidos
\ifcase #5 % c=0
\put{#4} at {\x0} {\y0}
\or % n=1
\put{\!pinta} at {\x0} {\y0}
\advance \y0 by \number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or % ne=2
\put{\!pinta} at {\x0} {\y0}
\advance \auxa by -2 % para fazer o ajuste (imperfeito)
\advance \auxb by -2 % ao raio da circunferencia de centro (x,y)
\advance \x0 by \number\auxa % width+5
\advance \y0 by \number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or % e=3
\put{\!pinta} at {\x0} {\y0}
\advance \x0 by \number\auxa % width+5
\put{#4} at {\x0} {\y0}
\or % se=4
\put{\!pinta} at {\x0} {\y0}
\advance \auxa by -2 % para fazer o ajuste (imperfeito)
\advance \auxb by -2 % ao raio da circunferencia de centro (x,y)
\advance \x0 by \number\auxa % width+5
\advance \y0 by -\number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or % s=5
\put{\!pinta} at {\x0} {\y0}
\advance \y0 by -\number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or % sw=6
\put{\!pinta} at {\x0} {\y0}
\advance \auxa by -2 % para fazer o ajuste (imperfeito)
\advance \auxb by -2 % ao raio da circunferencia de centro (x,y)
\advance \x0 by -\number\auxa % width+5
\advance \y0 by -\number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or % w=7
\put{\!pinta} at {\x0} {\y0}
\advance \x0 by -\number\auxa % width+5
\put{#4} at {\x0} {\y0}
\or % nw=8
\put{\!pinta} at {\x0} {\y0}
\advance \auxa by -2 % para fazer o ajuste (imperfeito)
\advance \auxb by -2 % ao raio da circunferencia de centro (x,y)
\advance \x0 by -\number\auxa % width+5
\advance \y0 by \number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\fi
\or % grafos nao dirigidos
\ifcase #5 % c=0
\put{#4} at {\x0} {\y0}
\or % n=1
\put{\!pinta} at {\x0} {\y0}
\advance \y0 by \number\auxb % height+depth+5

```

```

\put{#4} at {\x0} {\y0}
\or
% ne=2
\put{!pinta} at {\x0} {\y0}
\advance \auxa by -2 % para fazer o ajuste (imperfeito)
\advance \auxb by -2 % ao raio da circunferencia de centro (x,y)
\advance \x0 by \number\auxa % width+5
\advance \y0 by \number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or
% e=3
\put{!pinta} at {\x0} {\y0}
\advance \x0 by \number\auxa % width+5
\put{#4} at {\x0} {\y0}
\or
% se=4
\put{!pinta} at {\x0} {\y0}
\advance \auxa by -2 % ver acima
\advance \auxb by -2
\advance \x0 by \number\auxa % width+5
\advance \y0 by -\number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or
% s=5
\put{!pinta} at {\x0} {\y0}
\advance \y0 by -\number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or
% sw=6
\put{!pinta} at {\x0} {\y0}
\advance \auxa by -2 % ver acima
\advance \auxb by -2
\advance \x0 by -\number\auxa % width+5
\advance \y0 by -\number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\or
% w=7
\put{!pinta} at {\x0} {\y0}
\advance \x0 by -\number\auxa % width+5
\put{#4} at {\x0} {\y0}
\or
% nw=8
\put{!pinta} at {\x0} {\y0}
\advance \auxa by -2 % ver acima
\advance \auxb by -2
\advance \x0 by -\number\auxa % width+5
\advance \y0 by \number\auxb % height+depth+5
\put{#4} at {\x0} {\y0}
\fi
% \or % grafos dirigidos com molduras circulares nos objectos
% \advance \auxa by 4
% \put{\circle{\auxa}} [B1] at {\x0} {\y0}
% \put{#4} at {\x0} {\y0}
% \or % grafos não dirigidos com molduras circulares nos objectos
\else % grafos com molduras circulares nos objectos
\ifnum\auxa<\auxb % determina a maior das dimensões
\aux=\auxb
\else
\aux=\auxa
\fi
% se a largura da caixa é menor do que 1em então o tamanho
% tamanho é ajustado para esse valor mínimo
\ifdim\wd\caixa<1em
\dimen99 = 1em
\aux=\dimen99 \divide \aux by 131072
\advance \aux by 5
\fi
\advance\aux by -2 %folga entre o objecto e a moldura
\multiply\aux by 2 %
\ifnum\aux<30
\put{\circle{\aux}} [B1] at {\x0} {\y0}
\else
\multiply\auxa by 2
\multiply\auxb by 2
\put{\oval(\auxa,\auxb)} [B1] at {\x0} {\y0}
\fi
\put{#4} at {\x0} {\y0}
\fi
}

```

```
\catcode'\!=12 % ***** THIS MUST NEVER BE OMITTED (see PicTeX)
```