

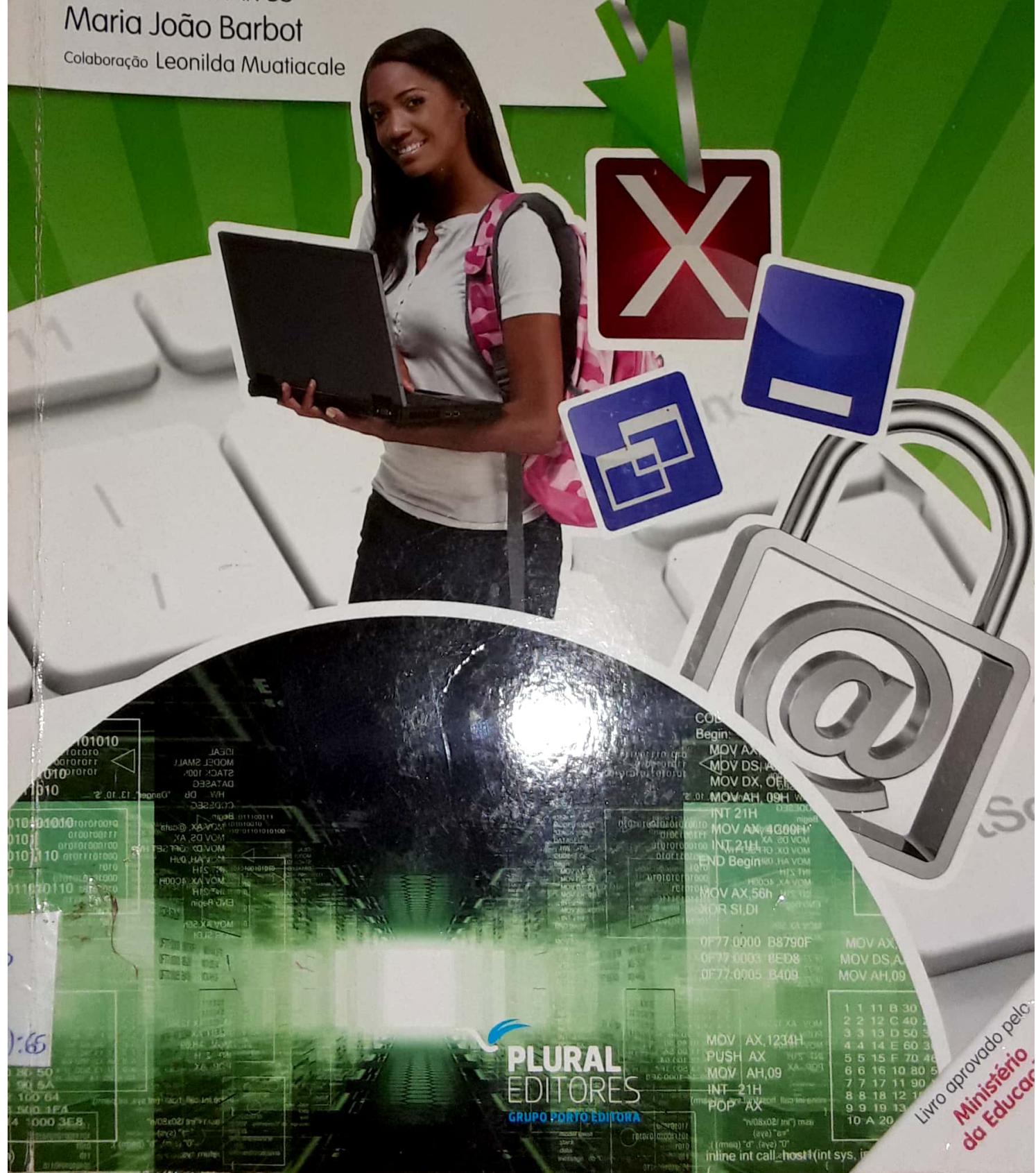
De acordo com os novos programas

TIC

Tecnologias de Informação e Comunicação

12^a classe

Maria Clara Alves
Maria João Barbot
Colaboração Leonilda Muatiacale



PLURAL
EDITORES
GRUPO PORTO EDITORA

```

MOV AX,1234H
PUSH AX
MOV AH,09
INT 21H
POP AX

```

MOV AH,09
 1 1 11 B 30
 2 2 12 C 40
 3 3 13 D 50
 4 4 14 E 60
 5 5 15 F 70
 6 6 16 10 80
 7 7 17 11 90
 8 8 18 12 100
 9 9 19 13 110
 10 A 20

30
40
50
60
70 46
80 5
90
100

Livro aprovado pelo
**Ministério
da Educação**

Título

TIC

Tecnologias de Informação e Comunicação

12.ª classe

Autoras

Maria Clara Alves

Maria João Barbot

Colaboração

Leonilda Muatiacale

Editora

Plural Editores

Copyright PORTO EDITORA



Moçambique

Av. Patrice Lumumba, 765 | Maputo

Telef. 21 36 09 00

Cel. 82 32 24 960 | 84 30 28 212

Fax 21 30 88 68

E-mail plural@pluraleditores.co.mz

www.pluraleditores.co.mz

2017

Execução gráfica Bloco Gráfico

PLD 8258/PLNLD/2014

12.ª
classe

TIC

Tecnologias de Informação e Comunicação

Cota 6- 8435

CDU 37(046-5):65

Maria Clara Alves

Maria João Barbot

Colaboração Leonilda Muatiacale

8435
19 07 2018
Cota Maria

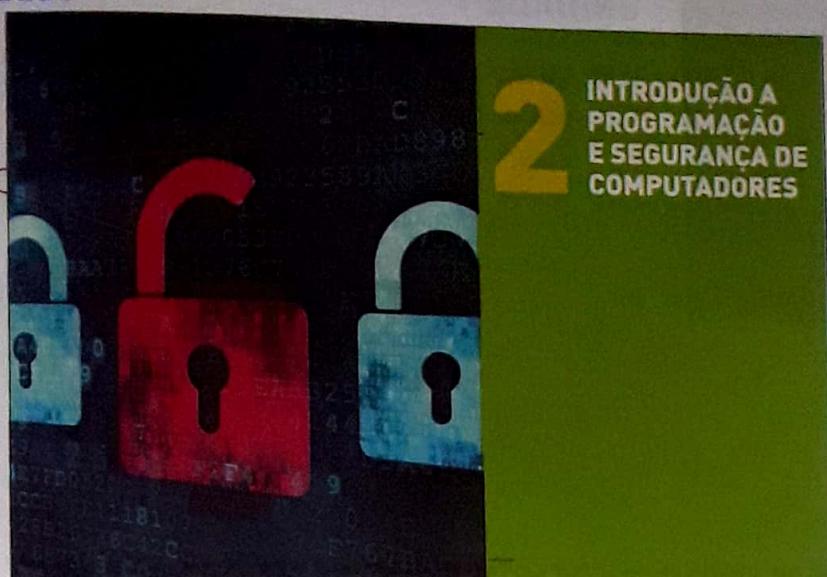
PLURAL
EDITORES
GRUPO PORTO EDITORA

8435

ESTRUTURA DO LIVRO

Abertura de unidade

Cada **unidade** abre em duas páginas com indicação do respectivo título e com uma imagem ilustrativa.



Desenvolvimento dos conteúdos

Quando um determinado passo de uma tarefa se encontra numerado a vermelho, significa que esse passo está ilustrado na imagem.

Final do livro

Existem os **Projectos**, com sugestões de trabalhos a efectuar ao longo do ano lectivo.

Cada unidade termina com propostas de **Actividades** de consolidação de conhecimentos.

Os **Apoios** visam auxiliar os alunos na realização das **Actividades** e **Projectos** relativos à unidade Trabalho de Projecto. No final do livro existe ainda um **Glossário**.

UNIDADE 1

Introdução de base de dados	8
1. Conceitos básicos	8
1.1. Conceito de base de dados	8
1.2. Sistema de Gestão de Bases de Dados (SGBD)	8
1.3. Terminologia das bases de dados	8
2. Organização de uma base de dados	13
3. Programa de gestão de bases de dados: Microsoft Office Access 2007	15
3.1. A janela do programa	16
3.2. O ambiente de trabalho	16
3.3. Elementos de uma base de dados	17
3.4. Criação de uma base de dados	18
3.5. Tabelas	20
3.6. Definição de chave primária	32
3.7. Relacionamento entre tabelas	33
3.8. Ordenação de registos	35
3.9. Impressão dos dados de uma tabela	35
3.10. Eliminação de tabelas	35
3.11. Consultas	36
3.12. Formulários	39
3.13. Relatórios	45
3.14. Macros	49
3.15. Conceito de módulo	51
4. Segurança da informação	52
5. Recuperação da informação	52
Actividades	53

UNIDADE 2

Introdução a programação e segurança de computadores

Introdução a programação	
1. Conceitos básicos	66
1.1. Sistema informático	66
1.2. Ambiente de programação	66
2. Algoritmos	66
2.1. Conceito	67
2.2. Construção de um algoritmo	67
3. Variáveis	68
4. Tipos de dados	68
5. Operadores	69
6. Expressões	70
7. Operações de atribuição	70
7.1. Compatibilidade de tipos	72
7.2. Funções embutidas	72
8. Comandos de entrada e saída	73
8.1. Comando de entrada	74
8.2. Comando de saída	74

9. Descrição de algoritmos	76
9.1. Descrição natural	76
9.2. Diagramas lógicos - fluxogramas	76
9.3. Linguagem algorítmica (pseudocódigo)	78
9.4. Traçagem de algoritmos	79
10. Estrutura de decisão	79
10.1. Estrutura de decisão simples	79
10.2. Estrutura de decisão encadeada	81
11. Estrutura de repetição (ciclos)	83
11.1. Ciclos condicionais (Repita enquanto)	83
11.2. Ciclos contadores (Repita para)	87
12. Vectores	90
12.1. Operações com vectores	90
12.2. Ordenação de vectores	93
12.3. Pesquisa em vectores	96
13. Matrizes	98
14. Sub-rotinas	100
14.1. Funções	100
14.2. Procedimentos	101
Actividades	102
 Segurança de computadores	
15. Segurança e gestão de antivírus	104
15.1. Vírus informáticos	104
15.2. Spyware	106
Actividades	109

UNIDADE 3

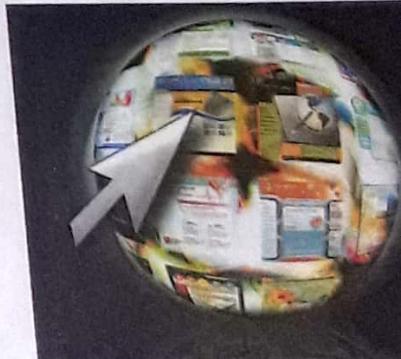
Trabalho de projeto

1. Conceito	112
2. Papel dos professores	112
3. Papel dos alunos	113
4. As etapas de desenvolvimento de um projeto	114
4.1. Identificar o problema	114
4.2. Dividir em unidades parcelares	115
4.3. Esboçar e planificar	116
4.4. Investigar e produzir	117
4.5. Apresentar trabalhos	118
4.6. Implementar e avaliar	119
Actividades	120

Projectos	128
------------------	-----

Apoios	130
---------------	-----

Glossário	144
------------------	-----





1

INTRODUÇÃO DE BASE DE DADOS

INTRODUÇÃO DE BASE DE DADOS

1. CONCEITOS BÁSICOS

1.1. Conceito de base de dados

Uma **base de dados** tem como objectivo o armazenamento de diversos tipos de informação relacionada, de tal forma que a sua actualização e consulta possam ser eficientes e efectuadas num curto espaço de tempo.

Pensando numa base de dados não informatizada, tens, por exemplo, todos os testes das diversas disciplinas que efectuaste ao longo do ano lectivo. Contudo, normalmente associamos o termo base de dados a grandes quantidades de informação e, sendo assim, poderás pensar, por exemplo, no arquivo existente na biblioteca da tua escola sobre livros e respectivos autores, ou no arquivo com os processos dos alunos, professores e funcionários. Como facilmente se percebe, encontrando-se esta informação em papel, a sua consulta pode ser demorada e levar a enganos.

1.2. Sistema de Gestão de Bases de Dados (SGBD)

Existem inúmeros programas específicos para a Gestão de Bases de Dados. Uns mais indicados em situações em que é necessário manipular um grande volume de informação (ex.: Informix Dynamic Server, Oracle Server, Sybase SQL Server, Microsoft SQL Server) e outros mais indicados para uso pessoal e/ou pequenas empresas (ex.: Microsoft Access e OpenOffice Base). O OpenOffice Base é um dos programas que constituem o OpenOffice e a sua transferência pode ser feita gratuitamente a partir da Internet. Contudo, devido às suas potencialidades e ao facto de ser um dos programas mais utilizados, neste livro vais aprender a utilizar o Microsoft Access 2007, que integra o Microsoft Office 2007.

O aparecimento de aplicações informáticas que permitem a introdução dos dados, o seu relacionamento e a posterior consulta, no ecrã ou impressa, provocou alterações significativas quer no suporte em que o registo é feito (o papel é substituído pelo registo digital), quer na fiabilidade dos resultados produzidos (minimização de erros), quer, ainda, no tempo útil em que a informação é disponibilizada. Surgem, assim, os **Sistemas de Gestão de Bases de Dados (SGBD)**.

Um **Sistema de Gestão de Bases de Dados (SGBD)** pode ser definido como um conjunto de aplicações que fazem a gestão e o processamento de uma grande quantidade de dados. Os **dados**, por sua vez, estão armazenados (guardados) de uma forma estruturada, permitindo um fácil acesso à informação, no formato mais adequado.

1.3. Terminologia das bases de dados

1.3.1. Noção de entidade e tabela

Definimos como **entidade** qualquer objecto, com características comuns, e que pode ser caracterizada por um conjunto de atributos de forma inequívoca.

Num **SGBD**, as entidades correspondem a **tabelas**, que são utilizadas para guardar informação (**dados**) relativa ao objecto.

Ex.: Num
fiche
defi

Transpo
Alunos, qu
Pensan
dade Livr
dente pod
nados co

M
Nome
Número

Fig.1

1.3.2. N

Supõ
dar o sa
encarreg
semelha

Tabela: A

Númer
1
2
3

1
Cada
à um
Um
dade
Respon
Nu e

Ex.: Numa escola, no acto da inscrição, os alunos devem preencher uma ficha com informações que os permita identificar. Assim, podemos definir a **entidade Aluno** especificando os seus **atributos** relevantes.

Transpondo para um SGBD, a entidade **Aluno** poderá dar origem à tabela **Alunos**, que deverá conter, unicamente, dados relacionados com os alunos.

Pensando nos livros existentes numa biblioteca, podemos definir a **entidade Livro** com os respectivos **atributos**. Num SGBD, a tabela correspondente poderá chamar-se **Livros** e deverá conter, unicamente, dados relacionados com os livros.

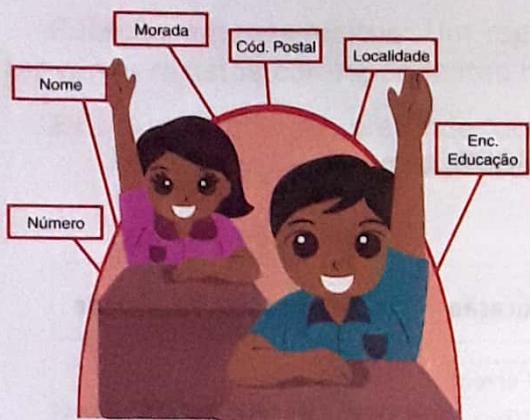


Fig. 1

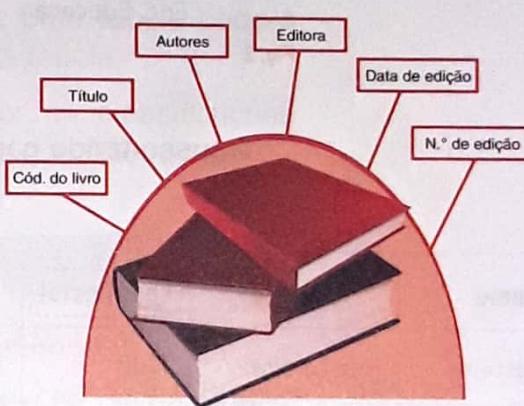


Fig. 2

1.3.2. Noção de registo, campo e dado

Supõe que, relativamente aos alunos de uma escola, se pretende guardar o seu número, nome, morada, código postal, localidade e nome do encarregado de educação (Fig. 1). Criando uma tabela obterias um resultado semelhante ao seguinte:

Tabela: Alunos

Número	Nome	Morada	Cód. Postal	Localidade	Enc. Educação
1	José Ferreira	Rua Direita	1100	Maputo 1	Paulina Ferreira
2	Noémia Santos	Rua Central	1101	Polana	Sérgio Santos
3	Reinaldo Guerra	Av. 24 de Julho	1100	Maputo 1	Maria Guerra

3

1 Cada linha da tabela corresponde a um **registo**.

Um registo caracteriza uma entidade concreta.

Repara que cada linha da tabela diz respeito a um, e apenas um, aluno.

No exemplo temos três registo e, cada um, caracteriza um aluno diferente.

2 Cada coluna corresponde a um **campo**.

Os campos especificam os diversos dados ou categorias que pretendemos especificar e que caracterizam a entidade.

No exemplo temos os campos **Número**, **Nome**, **Morada**, **Cód. Postal**, **Localidade** e **Enc. Educação**.

3 O conteúdo de cada campo é um **dado** e pode ser definido como um acontecimento ou um facto, sem tratamento adicional. Assim, um dado, quando isolado, não tem significado.

No exemplo existem diversos dados, nomeadamente, **1**, **Rua Direita**, **1100**, **Maputo 1**, etc.

3.4. Criação de uma base de dados

Utilizando o assistente de base de dados

Por este processo podes criar uma base de dados a partir de um modelo predefinido.

Um **modelo** consiste numa base de dados, relativa a um tema ou assunto específico, composta por tabelas, consultas, formulários e relatórios. O Access disponibiliza vários modelos que podes utilizar, adequando-os de modo a facilitar o processo de criação de bases de dados.

Supõe que pretendes criar uma base de dados de contactos.

Para tal:

1. Acede ao **Access**.
2. Na janela **Microsoft Access** clica em **Modelos Locais**.
3. Clica em **Contactos**.
4. Digita o nome que pretendes atribuir à base de dados.

Para alterar a localização onde a base de dados será criada, clica em e selecciona a *drive* e a pasta pretendidas.

5. Clica em **Criar**.

A respectiva base de dados é criada e é visualizada a tabela **Lista de Contactos**, na qual podes, desde logo, inserir dados.

6. Clica em cada um dos campos e digita o respectivo conteúdo.
7. Para inserir um novo **Contacto**, clica num campo da linha referenciada como **Novo**.



Fig. 9

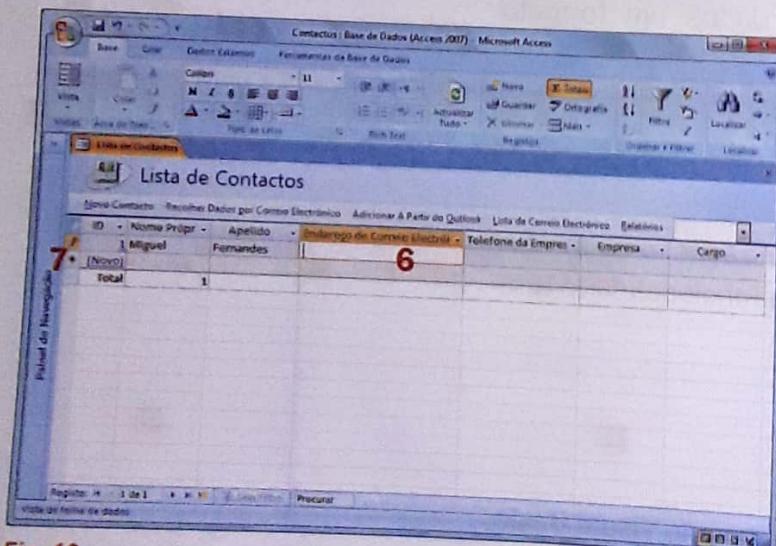


Fig. 10

Nome Próprio, Apelido, Endereço de Correio Electrónico, Telefone da Empresa, Empresa e Cargo são os campos que constituem cada **registo** da tabela **Lista de Contactos**. Logo que começas a escrever num campo, o registo é gravado.

Criar uma base de dados vazia

Ao criar uma base de dados vazia, nenhuma tabela, consulta, formulário ou relatório serão criados. Esta técnica é mais flexível, mas obriga-nos a definir todos os elementos que irão constituir a base de dados.

Para tal:

1. Acede ao Access.
2. Na janela Microsoft Access clica em **Base de Dados Vazia**.
3. Clica em  para seleccionares a pasta onde pretendes criar a base de dados.
4. Digita o nome a atribuir à base de dados.
5. Clica em **Criar**.

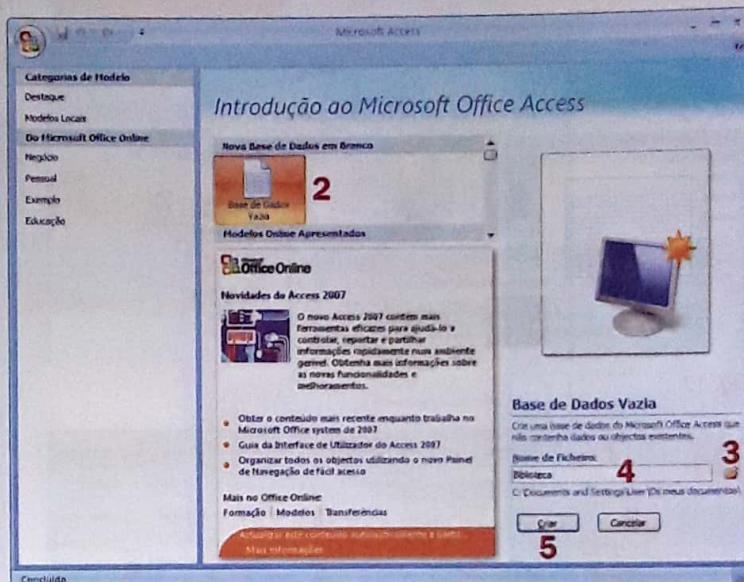


Fig. 11

A base de dados está pronta a ser utilizada e, desde logo, é sugerida a criação da primeira tabela.

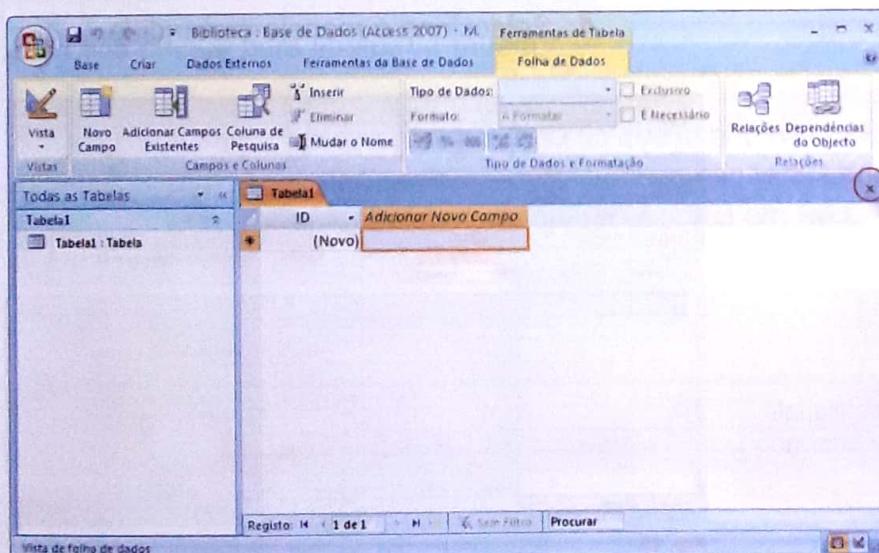


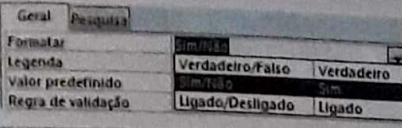
Fig. 12

- A base de dados fica, desde logo, gravada.
- O nome da base de dados é visualizado na **Barra de Título**.
- No **Painel de Navegação** é visualizada a tabela criada pelo Access.

Byte	Valores numéricos entre 0 e 255.
Número inteiro	Valores numéricos entre -32.768 e +32.767.
Número inteiro longo	Valores numéricos entre -2.147.483.648 e +2.147.483.647.
Simples	Valores numéricos com vírgula flutuante entre $-3,4 \times 10^{38}$ e $+3,4 \times 10^{38}$ e até sete dígitos significativos.
Duplo	Valores numéricos com vírgula flutuante entre $-1,797 \times 10^{308}$ e $+1,797 \times 10^{308}$ e até quinze dígitos significativos.
ID de replicação	Armazenamento de um identificador exclusivo global obrigatório para replicação.
Decimal	Valores numéricos entre $-9,999 \dots \times 10^{27}$ e $+9,999 \dots \times 10^{27}$.

Formatar

Tipo do campo	Descrição
Texto	<p>É possível definir um formato personalizado utilizando os símbolos:</p> <p>@ - obriga à introdução de pelo menos um carácter ou espaço;</p> <p>& - não obriga à introdução de caracteres;</p> <p>> - transforma todas as letras em maiúsculas;</p> <p>< - transforma todas as letras em minúsculas.</p>
Número	<p>Número geral - o número é visualizado tal como é introduzido.</p> <p>Moeda - o número é visualizado utilizando o separador de milhares. Os valores negativos, símbolos decimais e de moeda, bem como as casas decimais, respeitam as especificações das Definições Regionais.</p> <p>Euro - o número é visualizado com o símbolo de moeda do Euro.</p> <p>Fixo - é visualizado, pelo menos, um dígito e são aplicadas as especificações das Definições Regionais.</p> <p>Padrão - é visualizado o número utilizando o separador de milhares e aplicadas as especificações das Definições Regionais.</p> <p>Percentagem - multiplica o valor por 100 e apresenta o número com o sinal de percentagem.</p> <p>Científico - o número é visualizado na notação científica padrão.</p>
Data/Hora	<p>Data geral - é visualizada a data utilizando uma combinação das definições Data abreviada e Hora completa.</p> <p>Data por extenso - a data é visualizada utilizando a definição Data por extenso das Definições Regionais.</p> <p>Data normal - a data é visualizada no formato dd-mm-aa.</p> <p>Data abreviada - a data é visualizada utilizando a definição Data Abreviada das Definições Regionais.</p> <p>Hora completa - a hora é visualizada utilizando a definição Hora das Definições Regionais.</p> <p>Hora normal - a hora é visualizada no formato HH:MM PM, onde HH é a hora, MM são os minutos e PM é AM ou PM. A hora pode estar compreendida entre 1 e 12. Os minutos podem estar compreendidos entre 0 e 59.</p> <p>Hora abreviada - a hora é visualizada no formato HH:MM, onde HH é a hora e MM são os minutos. A hora pode estar compreendida entre 0 e 23. Os minutos podem estar compreendidos entre 0 e 59.</p>

Tipo do campo	Descrição
Sim/Não 	Verdadeiro/Falso – apresenta o valor como Verdadeiro ou Falso. Sim/Não – apresenta o valor como Sim ou Não. Ligado/Desligado – apresenta o valor como Ligado ou Desligado.

Relativamente a campos do tipo Texto analisa os exemplos:

Formatação	Texto digitado	Texto visualizado
@@@-@@@	4100001	4100-001
>	Matola-Rio MATOLA-rio	MATOLA-RIO
<	Matola-Rio MATOLA-rio	matola-rio

A formatação não altera o conteúdo do campo.

Máscaras de introdução

As máscaras de introdução têm como objectivo facilitar e controlar a introdução de dados num campo. A sua utilização é recomendada quando os dados devem respeitar um determinado tamanho e/ou formato, minimizando, assim, as possibilidades de erro.

Podes definir uma máscara de introdução directamente ou utilizar o Assistente.

Utilizando o Assistente:

1. Acede à base de dados a alterar.
2. Selecciona a tabela pretendida.
3. Clica no botão **Vista**.
4. Selecciona a opção **Vista de Estrutura**.

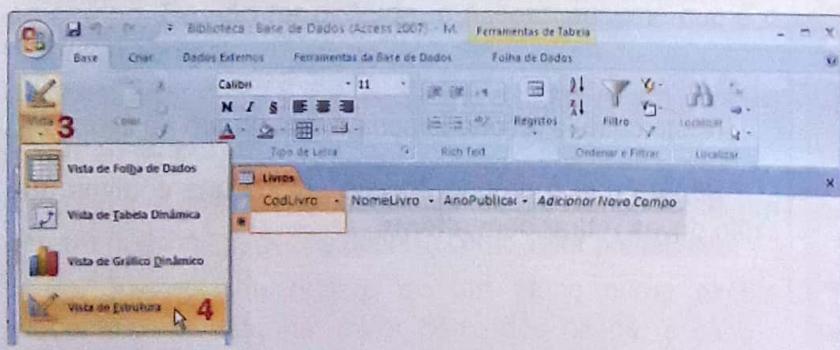


Fig. 24

5. Se o campo ainda não existir na tabela, indica o **Nome do campo**, o **Tipo de dados** e a **Descrição**.
6. Indica o **Tamanho** ou a **Formatação do campo**.
7. Clica no botão  para iniciar o **Assistente**.

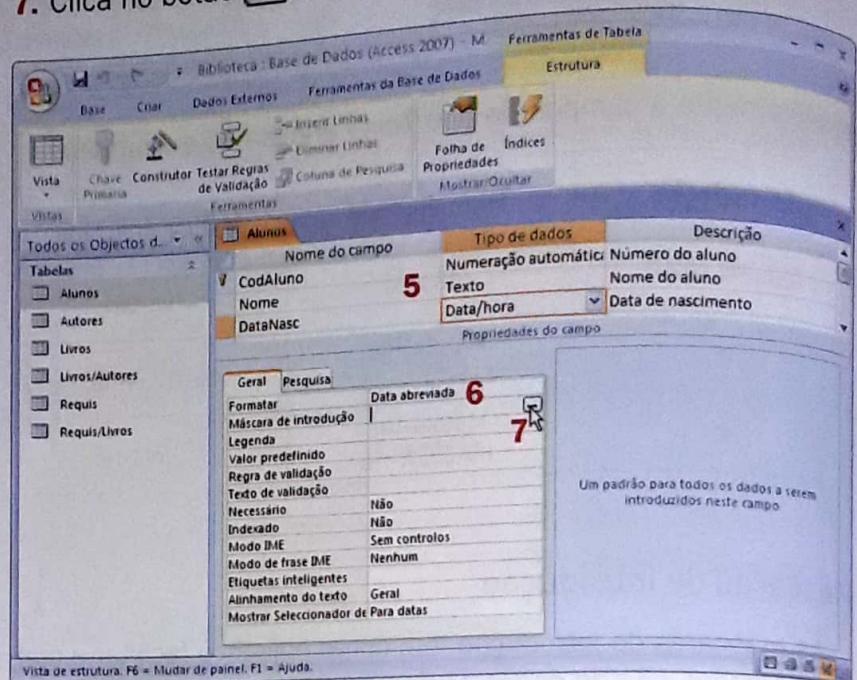


Fig. 25

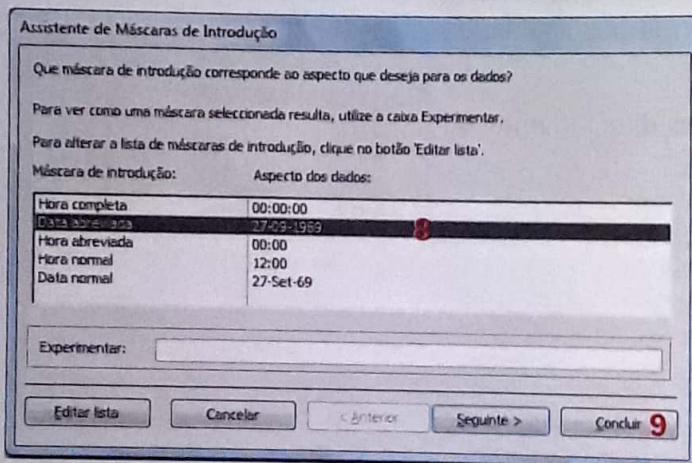


Fig. 26

8. Na janela **Assistente de Máscaras de Introdução** selecciona a opção mais adequada.

Clica na caixa de texto **Experimentar** para poderes testar como o campo será visualizado.

9. Clica em **Concluir**.

 Podes alterar o símbolo utilizado como marcador de posição. Para tal, clica em **Seguinte** e selecciona outro carácter.

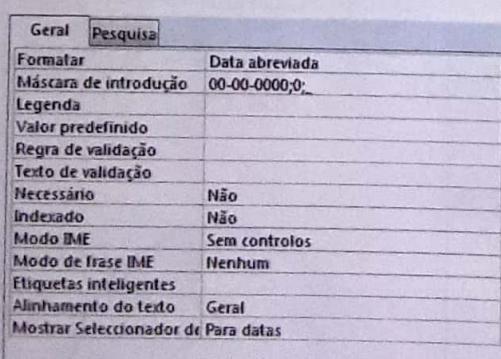


Fig. 27

Neste caso, o resultado seria o visualizado na figura 27. Repara que o próprio Access coloca uns códigos que não deves retirar nem alterar.

Definir a máscara de introdução de dados manualmente:

Para definires, manualmente, a máscara de introdução de dados, deves seguir os procedimentos atrás referidos até ao ponto 6.

Em seguida, em **Máscara de introdução** deves especificar a máscara a aplicar.

Na tabela seguinte são referidos os símbolos que podes utilizar para definir uma máscara de introdução.

Símbolo	Descrição
0	Dígito obrigatório (de 0 a 9).
9	Dígito não obrigatório (de 0 a 9).
#	Qualquer dígito ou espaço.
L	Letra obrigatória (de A a Z).
?	Letra opcional (de A a Z).
>	Converte os caracteres seguintes em maiúsculas.
<	Converte os caracteres seguintes em minúsculas.
A	Carácter alfanumérico (0 a 9 ou A a Z) obrigatório.
a	Carácter alfanumérico (0 a 9 ou a a z) opcional.
&	Carácter ou espaço obrigatórios.
C	Carácter ou espaço opcionais.
!	Obriga à introdução da direita para a esquerda.
Password	É visualizado um asterisco (*) por cada carácter digitado.

Podes definir um campo como **Texto** e especificar que apenas devem ser aceites números. Para tal, deves inserir na **Máscara de introdução** o símbolo **9**.

Legenda

A legenda corresponde ao título que pretendes atribuir ao campo. Se não for definida, o Access assume que é o nome do campo.

Valor predefinido

Quando é grande a probabilidade de um campo conter um determinado valor, podes defini-lo como **Valor predefinido**.

Por exemplo: o registo de um aluno numa escola moçambicana tem, na maior parte dos casos, o campo relativo ao Código do País preenchido com MZ.

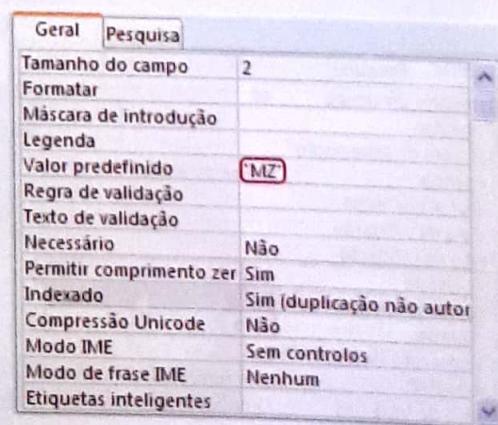


Fig. 28

O resultado final será semelhante ao seguinte:

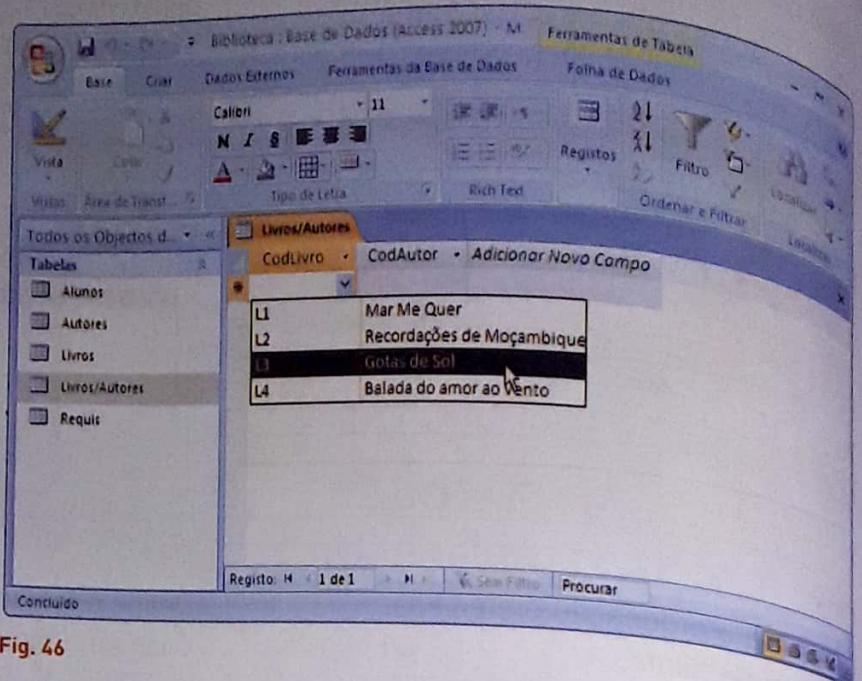


Fig. 46

Quando clicas na seta da caixa relativa ao campo **CodLivro** é visualizada a informação existente na tabela **Livros**.

Ao seleccionares o livro pretendido, o campo **CodLivro** da tabela **Livros/Autores** fica, de imediato, preenchido.

Claro que, antecipadamente, deves inserir informação na tabela **Livros**.

Tem todo o sentido repetires o procedimento para o campo **Autores**.

3.6. Definição de chave primária

Em cada tabela tem de existir pelo menos um campo, ou conjunto de campos, que permita(m) identificar, claramente, um, e apenas um, registo.

Por outras palavras, tem de haver pelo menos um campo, ou conjunto de campos, em que, por hipótese alguma, dois registos tenham o mesmo conteúdo. O campo, ou conjunto de campos, que satisfaça(m) este requisito constitui(em) a chave primária da tabela.

Considerando, por exemplo, uma tabela com informação sobre livros, ao definir o campo **CodLivro** como chave primária, estamos a impor que não poderão existir dois livros, na tabela, com o mesmo **CodLivro**.

Para definir campos como chave:

1. Selecciona a tabela.
2. No separador **Base**, activa a opção **Vista de Estrutura**.
3. Clica no campo que pretendes definir como chave.
4. No separador **Estrutura**, clica em  (Chave Primária).
5. Guarda a tabela.

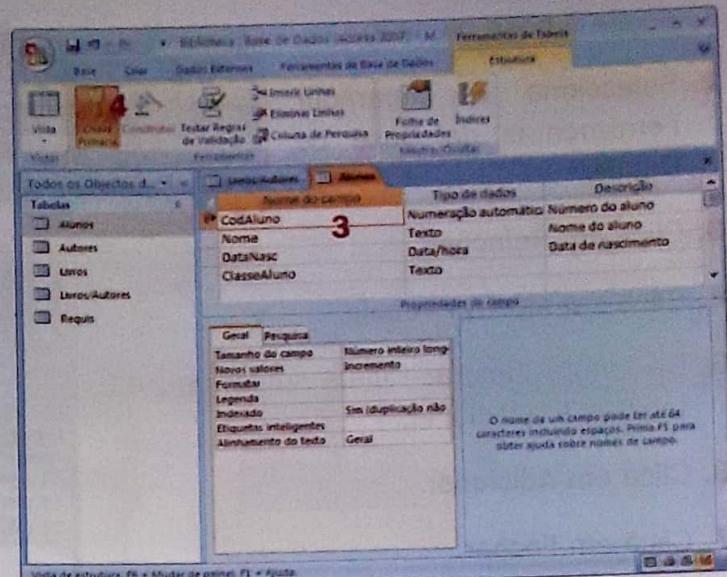


Fig. 47

Nome do campo	Tipo de dados	Descrição
CodAluno	Numeração automática	Número do aluno
Nome	Texto	Nome do aluno
DataNasc	Data/hora	Data de nascimento

Fig. 48

Há campos que não devem ser escolhidos como chave primária; é o caso dos nomes e apelidos. Isto porque, para além de, normalmente, conterem vários caracteres, há a possibilidade de se repetirem. É por este motivo que se recorre à definição de códigos e se definem estes campos como **campos-chave**.

3.7. Relacionamento entre tabelas

Para que possamos relacionar duas, ou mais, tabelas basta que o campo definido como chave primária de uma tabela esteja presente na outra tabela. Ou seja, as tabelas devem ter campos comuns.

Esta funcionalidade permite que, por exemplo:

- ao aceder à ficha de um aluno possamos visualizar, de imediato, a localidade onde ele reside. A **Localidade** é um campo da tabela **Códigos Postais**;
- ao aceder ao registo de classificações de um aluno possamos visualizar o nome de cada disciplina. O nome da **Disciplina** é um campo da tabela **Disciplinas**;
- ao aceder ao registo de requisição de um livro possamos visualizar o nome do aluno que o requisitou (informação da tabela **Alunos**), os autores do livro (informação da tabela **Autores**), etc.

Repara que do lado esquerdo do **CodAluno** aparece uma pequena chave indicando a funcionalidade do campo (Fig. 48).

Para definir mais do que um campo como chave, deves seleccionar os respectivos campos, mantendo a tecla **Ctrl** premida.

Os relacionamentos entre as tabelas devem ser definidos antes da introdução de dados, para evitar eventuais erros de inserção ou erros devidos a falhas na sua articulação.

1. Acede à base de dados.
2. Selecciona o separador **Ferramentas da Base de Dados**.
3. Clica em **Relações**.
4. Clica em **Mostrar Tabela**.
5. Na janela **Mostrar tabela** selecciona as tabelas a relacionar.
6. Clica em **Adicionar**.
7. Clica em **Fechar**.
8. Coloca o cursor sobre o campo a relacionar, prime o botão esquerdo do rato e arrasta o cursor até ao campo correspondente da outra tabela.
9. Na janela **Editar relações** clica em **Criar**.
10. Clica em  para guardar o esquema de relações.

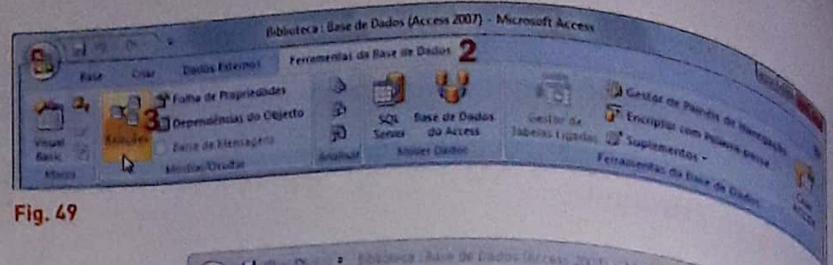


Fig. 49

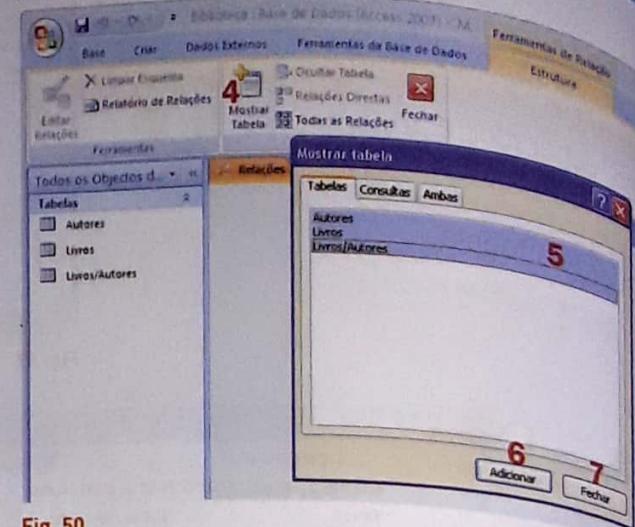


Fig. 50

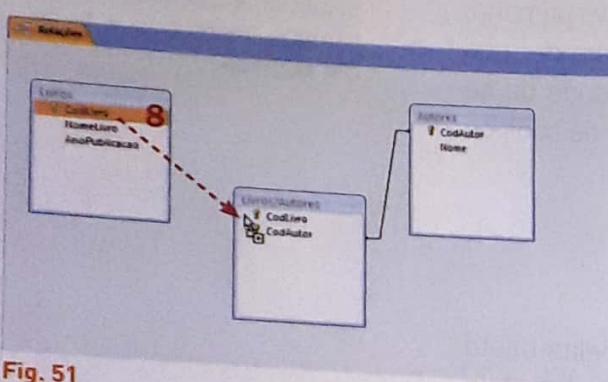


Fig. 51

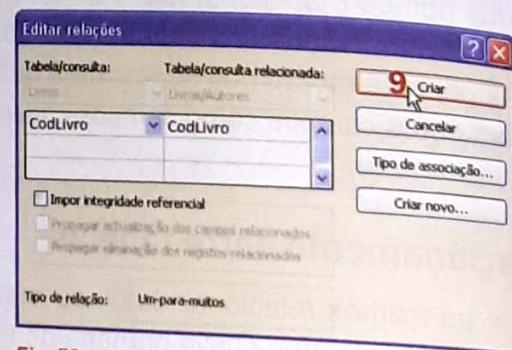


Fig. 52

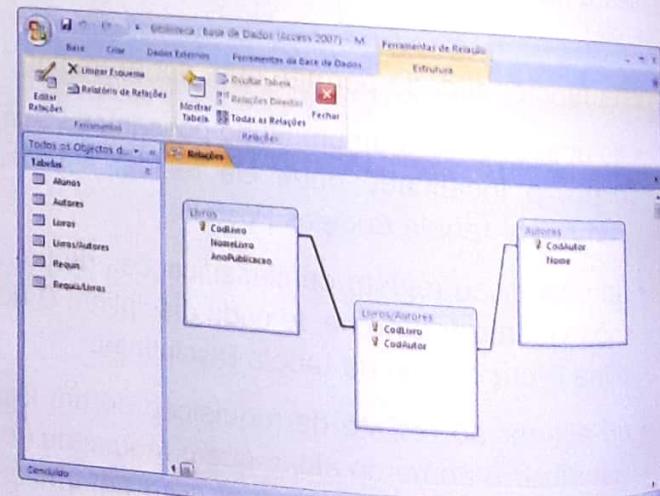


Fig. 53

O resultado final será semelhante ao da figura 53.

Clica em  para fechar o separador **Relações**.

3.8. Ordenação de registos

A inserção de dados nas tabelas não obedece a uma ordem específica. Contudo, podemos efectuar a sua ordenação, sempre que necessário, com base nos campos que a constituem.

1. Acede à base de dados.
2. Selecciona a tabela.
3. Clica no nome da coluna pela qual pretendes ordenar a informação.
4. No separador Base, grupo Ordenar e Filtrar, clica no botão correspondente à ordenação pretendida ($\downarrow\uparrow$ ou $\uparrow\downarrow$).

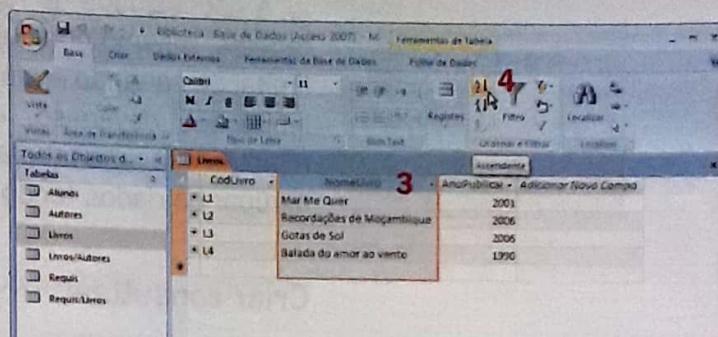


Fig. 54

3.9. Impressão dos dados de uma tabela

1. Acede à base de dados.
2. Selecciona a tabela cuja informação pretendes imprimir.
3. Clica no botão do Office.
4. Selecciona a opção Imprimir.
5. Selecciona as opções adequadas.
6. Clica em OK.

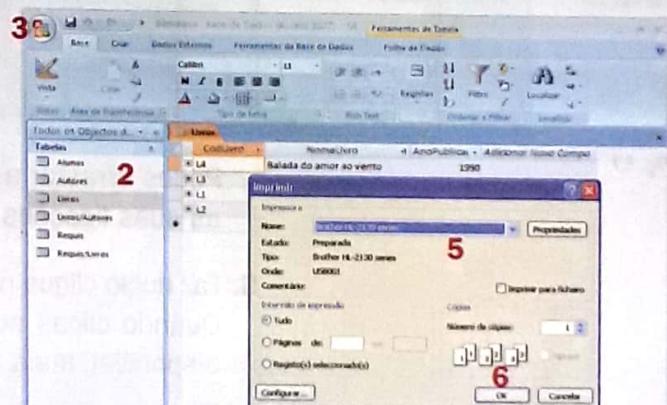


Fig. 55

3.10. Eliminação de tabelas

Para eliminar uma tabela, clica sobre ela, prime a tecla Del e, em seguida, confirma a eliminação (Fig. 56).

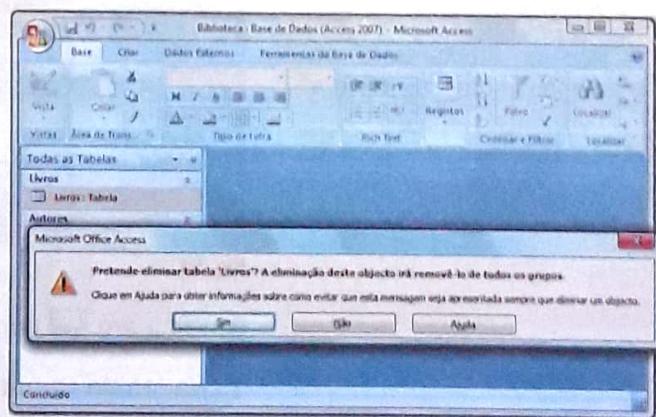


Fig. 56

3.11. Consultas

Uma consulta é um objecto do Access que permite localizar, seleccionar e relacionar registos de diversas tabelas que façam determinados critérios.

A necessidade de recorrer às consultas surge do facto de as tabelas, mesmo que bem estruturadas, apresentarem bastantes limitações quando pretendemos analisar a sua informação. Estas limitações devem-se, essencialmente, ao grande número de registos que uma tabela pode conter e, também, à dificuldade de agrupar os dados tendo em atenção determinados critérios.

Criar consultas

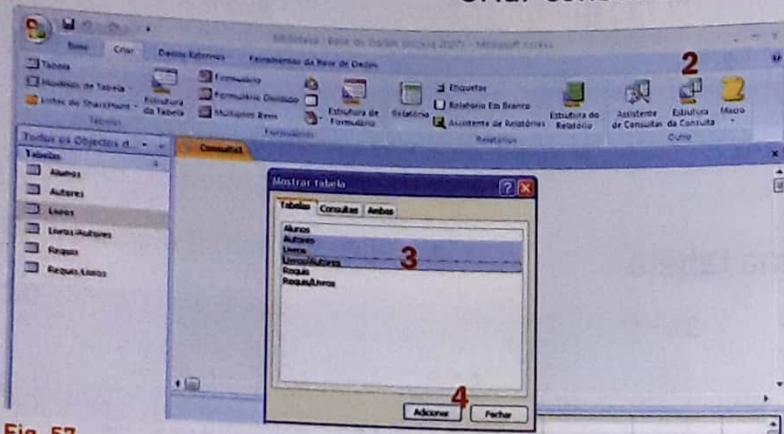


Fig. 57

1. Acede à base de dados.
2. Selecciona no separador Criar, grupo **Outro**, a opção **Estrutura da Consulta**.
3. Na janela **Mostrar tabela**, selecciona as tabelas que estejam envolvidas na consulta.
4. Clica em **Adicionar** e, em seguida, em **Fechar**.

Podes arrastar as tabelas para que possas visualizar melhor as suas ligações.

5. Faz duplo clique nos campos que pretendas incluir na consulta. Quando clicas num campo, ele vai ocupar a primeira coluna disponível, mais à direita.

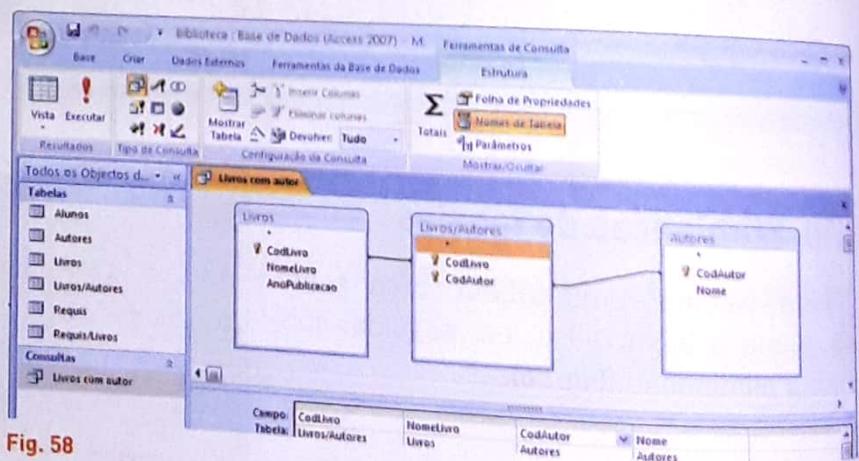


Fig. 58

 Na consulta representada na figura 58 pretende-se visualizar o nome dos livros e dos seus respectivos autores.

Analisa atentamente os campos que constituem a consulta.

6. Clica em  para guardar a consulta.
7. Na janela **Guardar como**, digita o nome da consulta.
8. Clica em **OK**.

Para visualizares de imediato a consulta, no separador **Estrutura**, clica em  (Executar).

Clica em  para fechar a consulta.

Para acederes novamente à estrutura da consulta, clica no botão **Vista** e selecciona a opção **Vista de Estrutura** (Fig. 59).

Executar uma consulta

1. Acede à base de dados.
2. Faz duplo clique sobre o nome da consulta que pretendes executar.

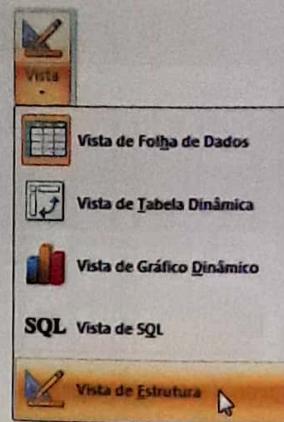


Fig. 59

Código do livro	Nome	Código do autor	Nome
L1	Mar Me Quer	A1	Mia Couto
L2	Recordações de Moçambique	A2	Joaquim Carlos Viegas
L2	Recordações de Moçambique	A3	Ana Paula Lemos
L3	Gotas de Sol	A4	Calane da Silva
L4	Balada do amor ao vento	A5	Paulina Chiziane

Fig. 60

Alterar a posição dos campos

1. Acede à consulta a que pretendes alterar a estrutura.
2. No separador **Base**, clica no botão **Vista** e selecciona a opção **Vista de Estrutura**.
3. Clica no nome da coluna a deslocar.
4. Prime o botão esquerdo do rato e, sem o largar, arrasta o cursor para a posição onde pretendes visualizar a informação.

Na figura 61, a coluna **Nome** é colocada à esquerda da coluna **CodAutor**.

Fig. 61

Se pretenderes que um campo continue a fazer parte da consulta mas não seja visível, desactiva a caixa de selecção respectiva, situada na linha **Mostrar**.

Eliminar um campo da consulta

A eliminação de um campo da consulta implica, apenas, que ele deixe de a constituir, não havendo qualquer repercussão na respectiva tabela. Para eliminar um campo, basta seleccionar a coluna que lhe diz respeito e premir a tecla Del.

Consultas com especificação de critérios

A especificação de critérios permite restringir os registos que uma consulta possa localizar: por exemplo, apenas os alunos do sexo masculino ou feminino ou que residam numa determinada localidade.

1. Seleciona a consulta.
2. No separador **Base**, clica no botão **Vista** e selecciona a opção **Vista de Estrutura**.
3. Na coluna pretendida e na linha correspondente ao **Critério**, digita o critério a aplicar.
4. Clica em  para guardar a consulta.
5. Clica em  para a executar.

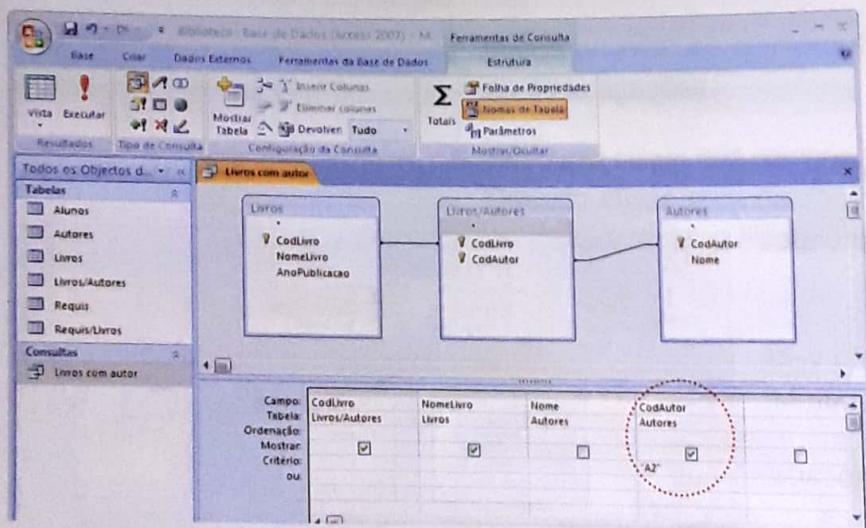


Fig. 62

Em cada uma das células da linha **Critério** podes escrever expressões utilizando operadores de comparação, aritméticos e lógicos.

Analisa o exemplo:

Especificado este critério ao executar a consulta se visualizados todos os livros do autor cujo código é **(Fig. 62)**.

Podes, também, utilizar o **Construtor de expressões**. Para

1. Clica na célula **Critério** do campo a que pretendes aplicar critério.
2. No separador **Estrutura** do grupo **Configuração da Consulta**, clica em **Construtor**.

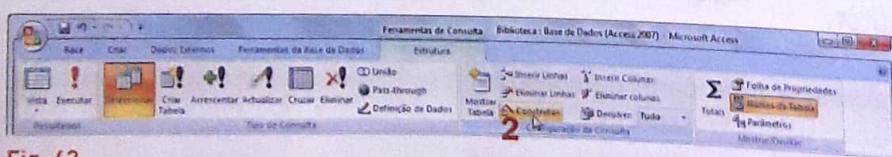


Fig. 63

1 **Caixa de expressão** – área onde podes escrever ou, recorrendo às secções inferiores, construir uma expressão.

2 **Botões de operador** – operadores que podes utilizar.

3 **Lista de pastas** – objectos da base de dados dos quais podes utilizar dados ou informação.

4 **Lista de elementos relativa à pasta seleccionada em 3.**

5 **Lista de valores, caso existam, relativos à pasta e elementos seleccionados.**

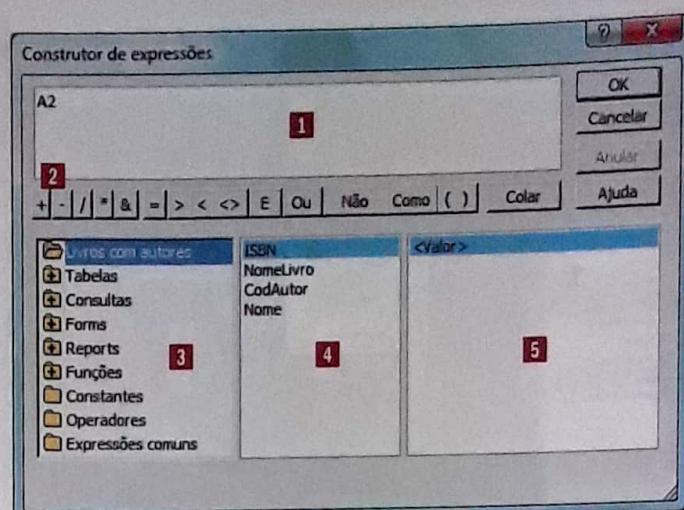


Fig. 64

Consultas com realização de cálculos

Para além dos campos das tabelas, numa consulta também podem existir colunas que expressem o resultado de cálculos.

1. No separador **Estrutura**, grupo **Mostrar/Ocultar**, clica em **Total**.

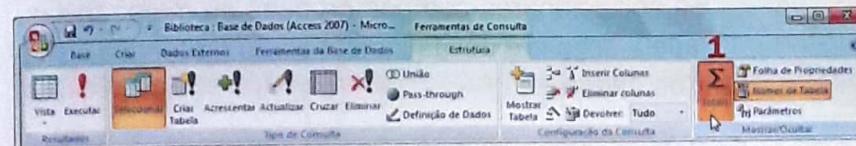


Fig. 65

2. Na linha correspondente ao **Total**, clica na seta da caixa de selecção e selecciona **Agrupar por**, nos campos que pretendes agrupar, ou a função/expressão a utilizar.

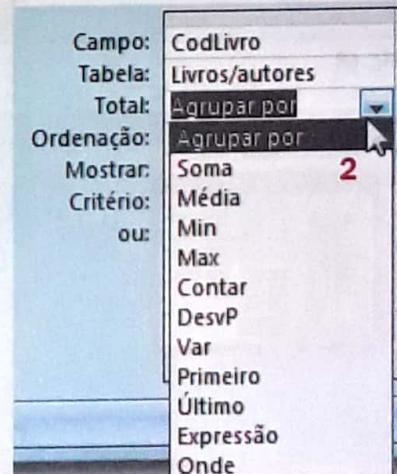


Fig. 66

3.12. Formulários

Fundamentalmente, os **formulários** são objectos do Access que facilitam a introdução, modificação, consulta e eliminação de dados nas tabelas. Podem também ser utilizados noutras situações, tais como: menus, janelas de configuração, janelas de diálogo, etc.

Na criação de um formulário há informação que provém da base de dados (dados existentes nas tabelas e/ou consultas) e outra que é inserida aquando do desenho do mesmo, sendo, portanto, constante (nome do formulário, etiquetas dos campos, botões de navegação, etc.).

Criar formulários utilizando o assistente

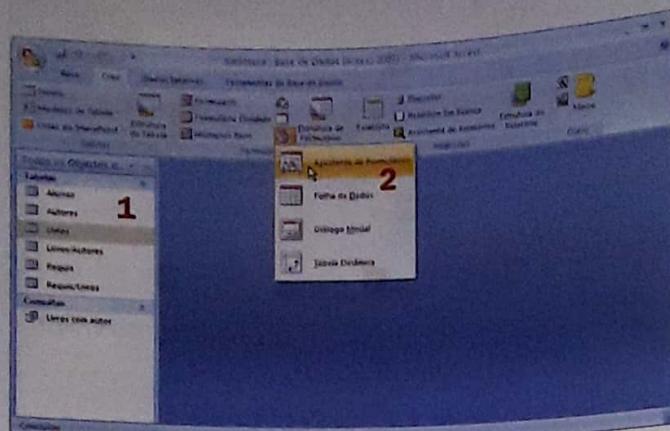


Fig. 67

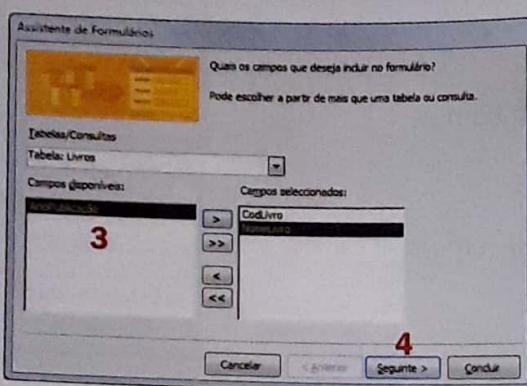


Fig. 68

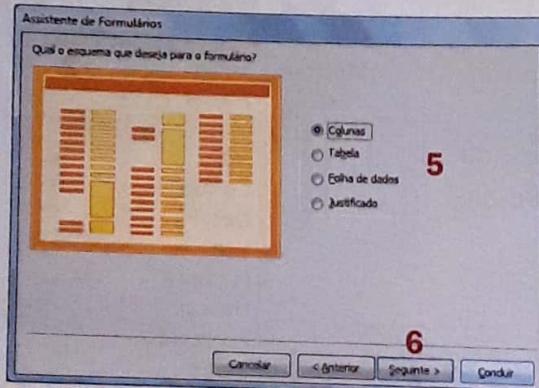


Fig. 69

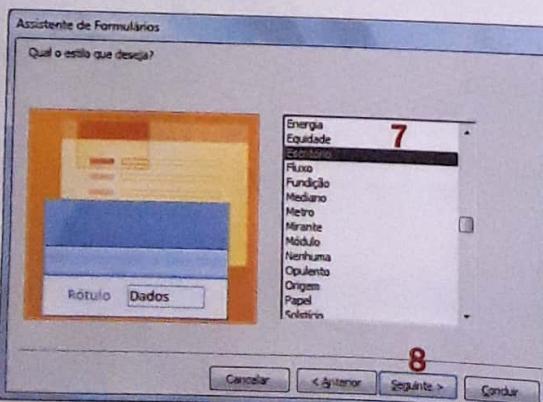


Fig. 70

1. Seleciona a tabela para a qual pretendes criar o formulário.
2. No separador Criar, grupo Formulários, clica em Mais Formulários e, em seguida, em Assistente de Formulários.
3. Faz duplo clique no campo para o seleccionares (passando para Campos selecionados). Ou clica no nome do campo e, em seguida, clica em >.
4. Clica em Seguinte.

Se pretenderes incluir campos de outras tabelas, desde que devidamente relacionadas, clica na seta da caixa de selecção Tabelas/Consultas para seleccionares a tabela. Os campos que constituem a tabela são visualizados na área Campos disponíveis e podes, então, seleccioná-los (Fig. 68).

Se seleccionares campos de duas ou mais tabelas, na janela seguinte deves indicar como pretendes visualizar os dados.

5. Seleciona o esquema que julgues mais adequado.
6. Clica em Seguinte.
7. Seleciona um estilo.
8. Clica em Seguinte.
9. O nome da tabela é sugerido para nome do formulário, mas podes alterá-lo.
10. Clica em Concluir.

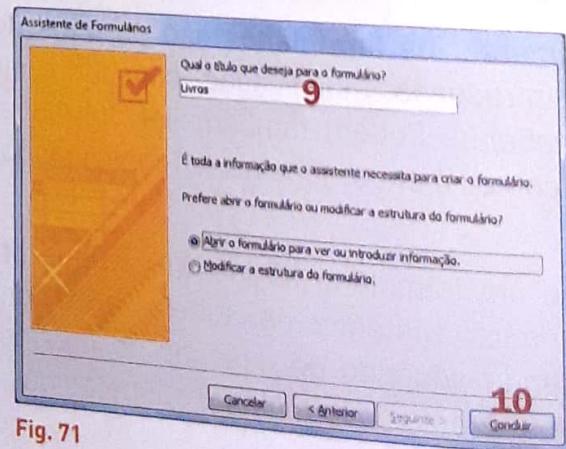


Fig. 71

Depois de concluído, o formulário é aberto. Podes agora visualizar, alterar e adicionar informação às tabelas (Fig. 72).

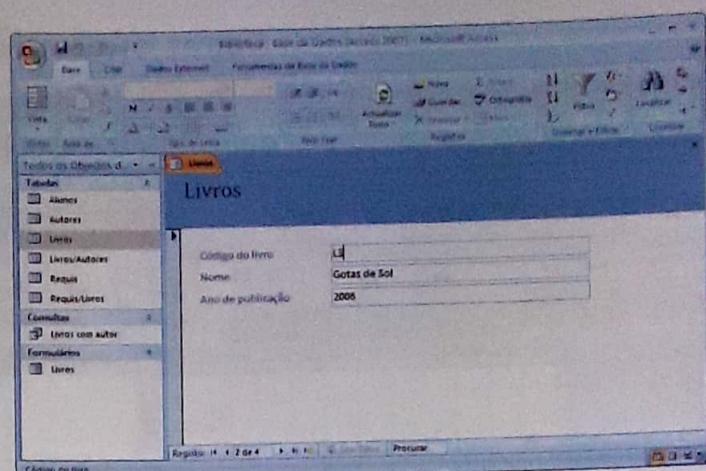


Fig. 72

Mover campos

Para movimentar os campos é necessário remover o esquema predefinido. Para tal:

1. Acede ao formulário em **Vista de Estrutura**.
2. Selecciona os campos.
3. Prime o botão direito do rato e, no menu de acesso rápido, selecciona a opção **Esquema**.
4. Selecciona a opção **Remover**.
5. Clica no campo a mover e arrasta-o para a posição pretendida.

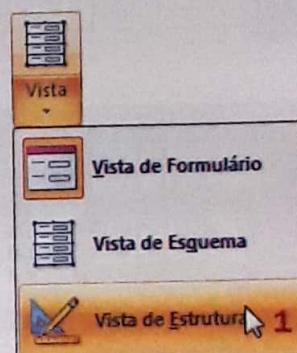


Fig. 73

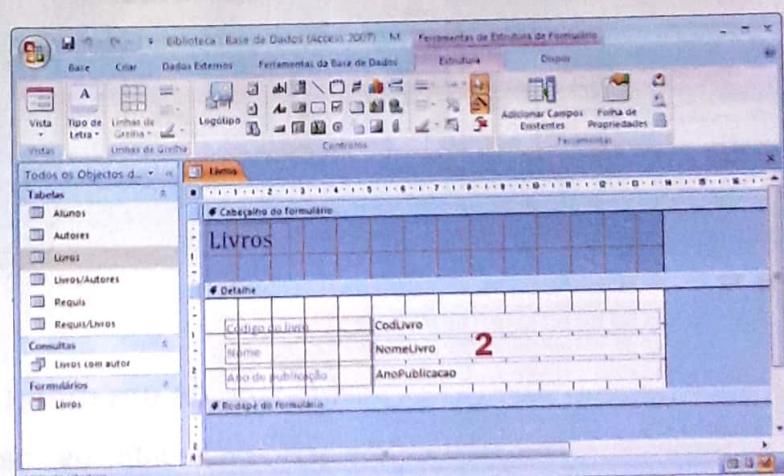


Fig. 74

Eliminar campos

Clica no campo a eliminar e prime a tecla **Del**.

Alinhar campos

1. Selecciona os campos a alinhar.
2. Prime o botão direito do rato e, no menu de acesso rápido, selecciona a opção **Alinhar**.
3. Selecciona o alinhamento pretendido.

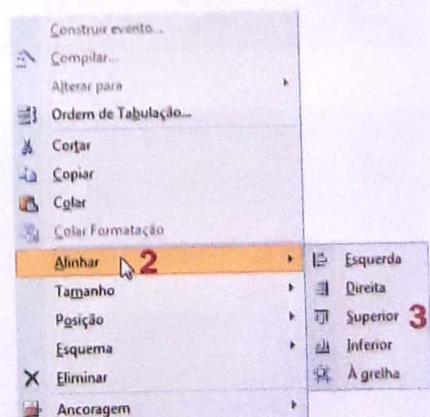


Fig. 75

Outras formatações

Depois de seleccionares o(s) campo(s), analisa as opções de formatação disponíveis no separador **Estrutura**.

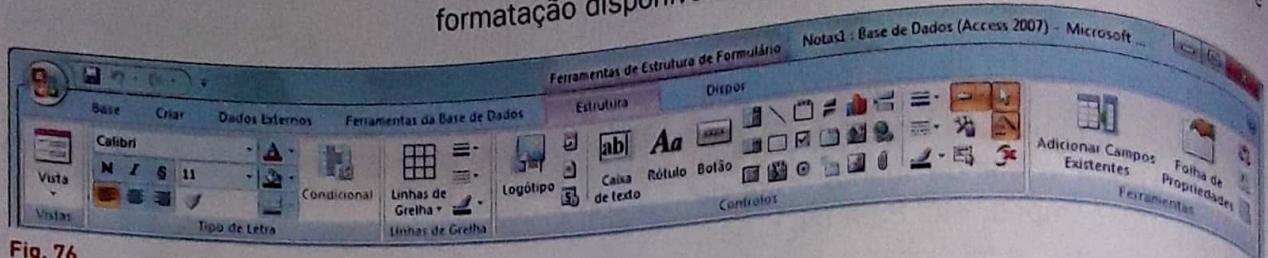


Fig. 76

Guardar um formulário

Para guardar as alterações efectuadas num formulário basta clicar em .

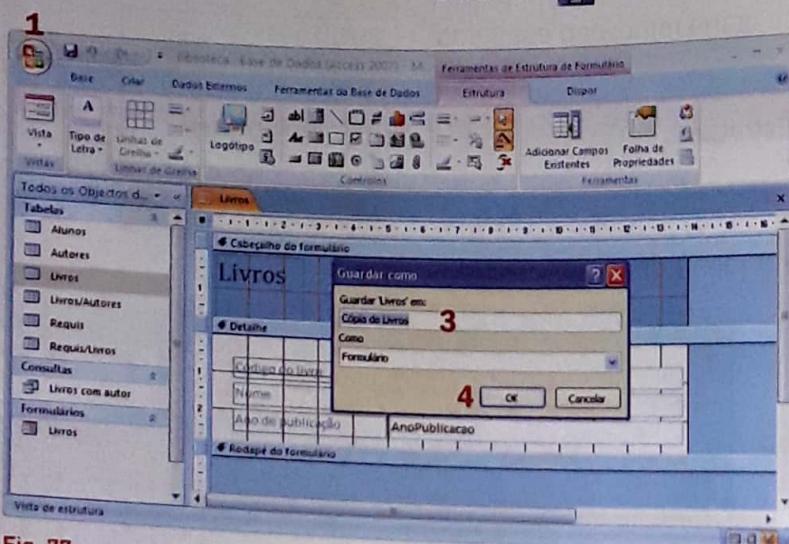


Fig. 77

Para guardar o formulário com outro nome:

1. Clica no botão do Office.
2. Selecciona a opção **Guardar como**.
3. Digita o nome que pretendes atribuir ao formulário.
4. Clica em **OK**.

Criação de um menu

Normalmente, os menus são utilizados para, rapidamente, aceder a formulários e consultas, imprimir relatórios, aceder a aplicações externas ao Access, etc.

Efectivamente, um menu não é mais do que um formulário constituído, essencialmente, por botões de comando que executam determinadas acções.

1. No separador **Criar**, grupo **Formulários**, clica em **Estrutura de Formulário**.

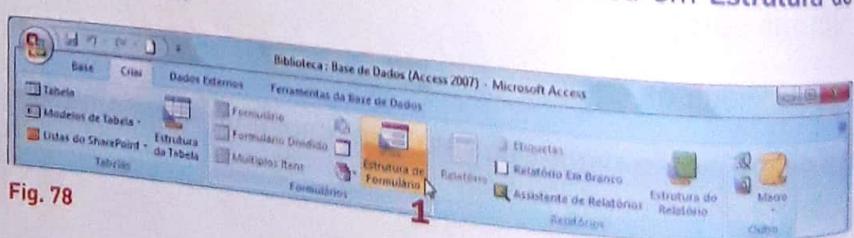


Fig. 78

Inserir um título no menu:

2. No menu **Ferramentas de Estrutura de Formulário**, separador **Estrutura**, grupo **Controlos**, clica no controlo **Rótulo**.

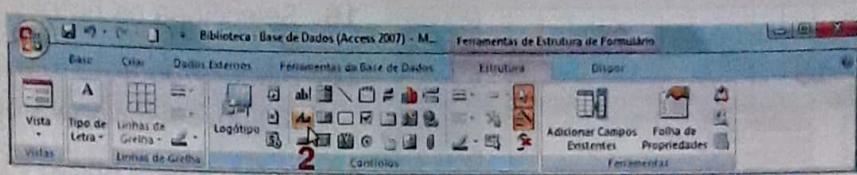


Fig. 79

3. Define uma área e digita o texto que pretendes como título.

Ajusta o tamanho da caixa e formata o texto a teu gosto.

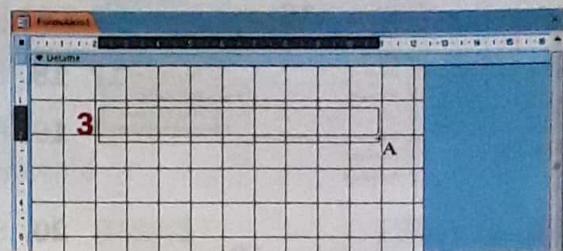


Fig. 80

Inserir botões de acesso a formulários:

4. No menu **Ferramentas de Estrutura de Formulário**, separador **Estrutura**, grupo **Controlos**, clica no controlo **Botão**.



Fig. 81

5. Define uma área para o botão.

6. Na janela **Assistente de Botões de Comando** selecciona a opção **Operações de formulários**.

7. Seleciona a acção **Abrir formulário**.

8. Clica em **Seguinte**.

9. Seleciona o formulário a abrir e clica em **Seguinte**.

10. Seleciona a opção adequada e clica em **Seguinte** (Fig. 83).

11. Clica no botão de selecção **Texto** e digita o nome do formulário a aceder.

12. Clica em **Seguinte**.

13. Clica em **Concluir**.

14. Ajusta o tamanho do botão.

Repete o procedimento para todos os formulários a que pretendas aceder.

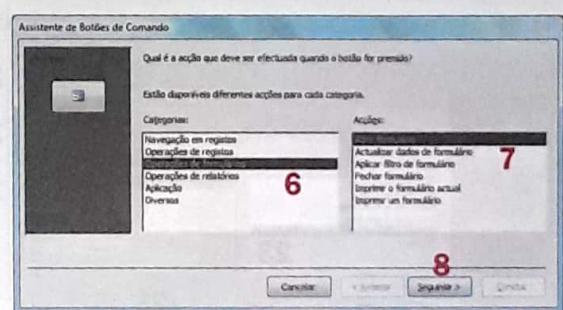


Fig. 82

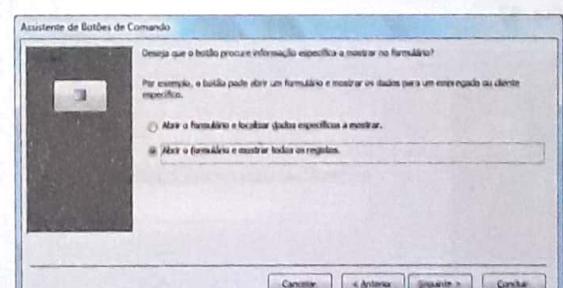


Fig. 83

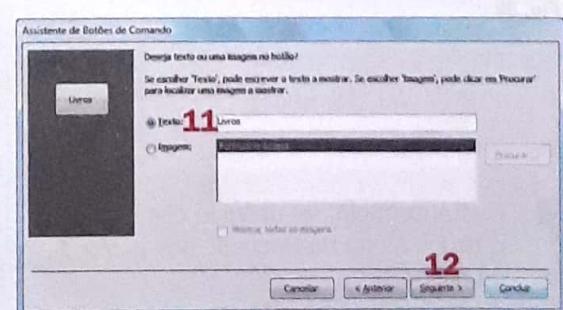


Fig. 84

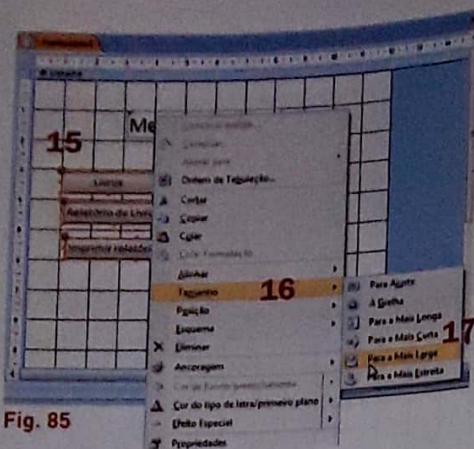


Fig. 85



Fig. 86

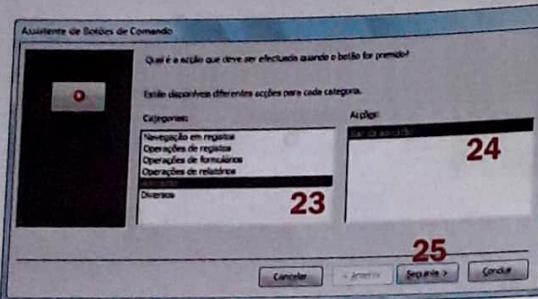


Fig. 87

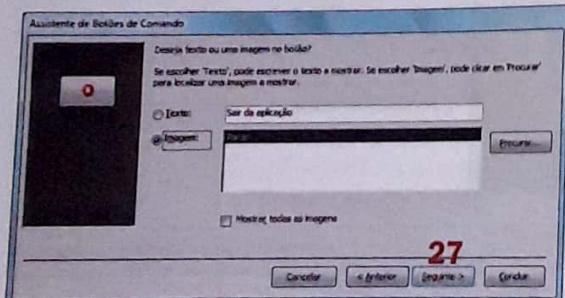


Fig. 88

Neste exemplo, foi definido como título o texto “Menu de Opções” (ver ponto 3 da pág. 43).

Alterar o tamanho dos botões:

15. Selecciona os botões.
16. Prime o botão direito do rato e, no menu de acesso rápido, selecciona a opção **Tamanho**.
17. Selecciona a opção **Para a Mais Larga**.

Alinhar os botões:

18. Selecciona os botões.
19. Prime o botão direito do rato e, no menu de acesso rápido, selecciona a opção **Alinhar**.
20. Selecciona a opção adequada.

Inserir um botão para abandonar o menu e sair do Access:

1. No menu **Ferramentas** de **Estrutura de Formulário**, separador **Estrutura**, grupo **Controlos**, clica no controlo **Botão**.
2. Define uma área para o botão.
3. Na janela **Assistente de Botões de Comando** selecciona a opção **Aplicação**.
4. Selecciona a acção **Sair da aplicação**.

24. Selecciona a acção **Sair da aplicação**.
25. Clica em **Seguinte**.

26. Clica no botão de selecção **Imagen**.

26. Clica no botão **Procurar**. Podes optar por utilizar outra imagem. Para tal basta clicar no botão **Procurar** e indicar a localização e o nome da imagem pretendida (Fig. 88).

27. Clica em **Seguinte**.

28. Clica em **Concluir**.

29. Guarda o formulário com um nome sugestivo. Deverás obter um formulário semelhante ao da figura 8.

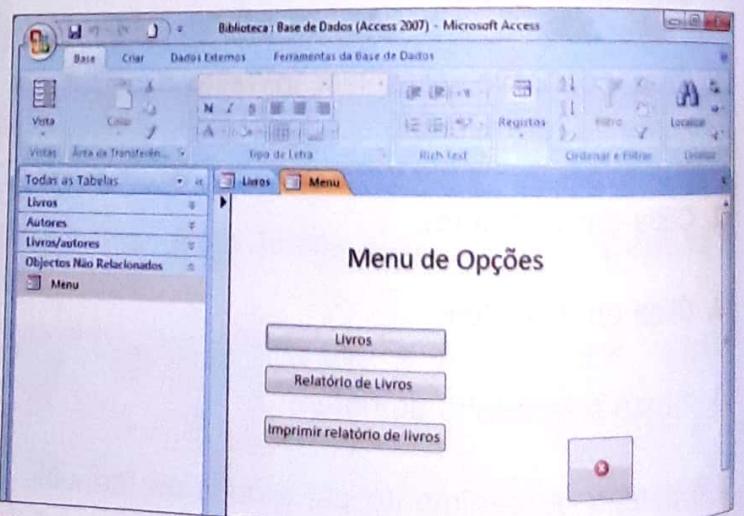


Fig. 89

3.13. Relatórios

Os relatórios são uma ferramenta do Access que permite, de forma eficaz e sugestiva, apresentar os dados existentes em tabelas e/ou consultas, num formato impresso.

A diferença entre os formulários e os relatórios é que estes não permitem a introdução, alteração ou eliminação de informação. Mas, por outro lado, permitem agregar os dados para que possam conter totais, subtotais e sumários dos dados listados.

1. Clica na tabela da qual pretendes obter o relatório.
2. No separador Criar, grupo Relatórios, clica em Assistente de Relatórios.
3. Faz duplo clique nos campos a incluir no relatório (passando-os para Campos selecionados).
4. Ou clica sobre o campo e, em seguida, clica em .
4. Clica em Seguinte.

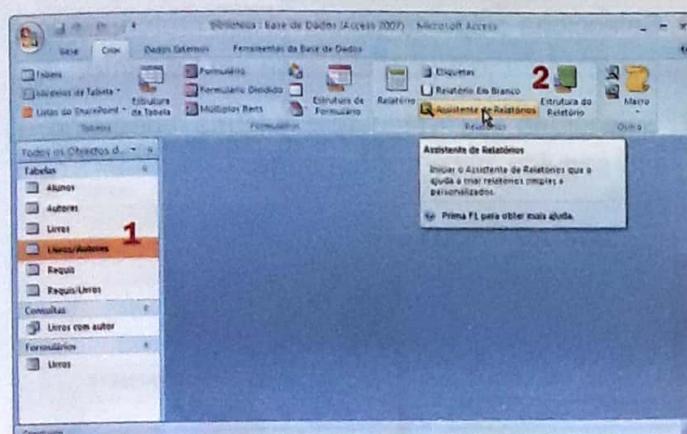


Fig. 90

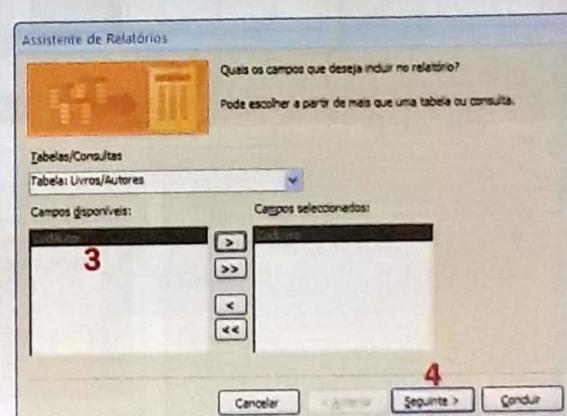


Fig. 91

Se pretenderes incluir campos de outras tabelas, desde que devidamente relacionadas, clica na seta da caixa de seleção **Tabelas/Consultas** para seleccionares a tabela (Fig. 92). Os campos que constituem a tabela são visualizados na área **Campos disponíveis** e podes, então, seleccioná-los.

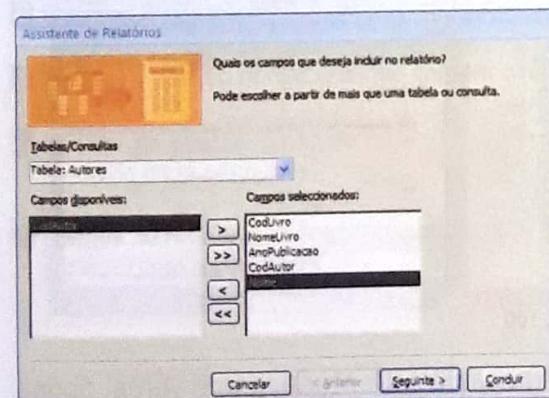


Fig. 93

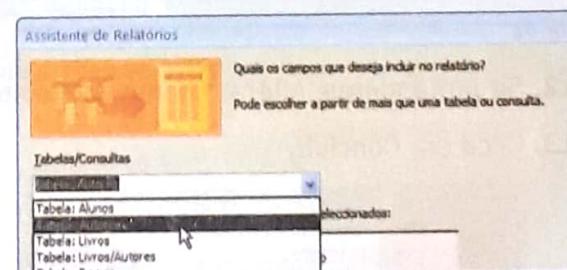


Fig. 92

Tendo em atenção a base de dados, com informação de livros, que tem sido utilizada como exemplo, na figura 93 foram seleccionados os campos:

- CodLivro, da tabela Livros/Autores;
- NomeLivro, da tabela Livros;
- AnoPublicacao, da tabela Livros;
- CodAutor, da tabela Livros/Autores;
- Nome, da tabela Autores.

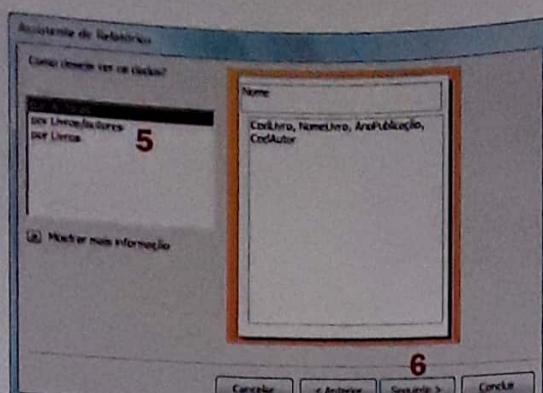


Fig. 94

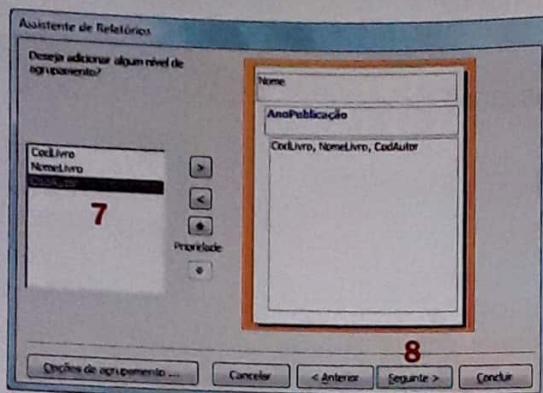


Fig. 95

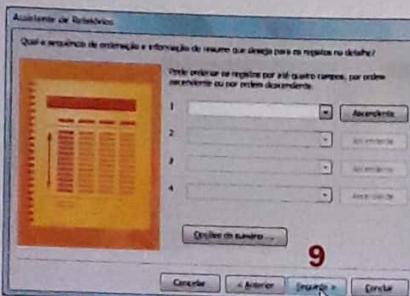


Fig. 96

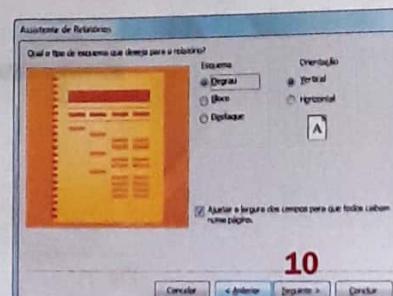


Fig. 97

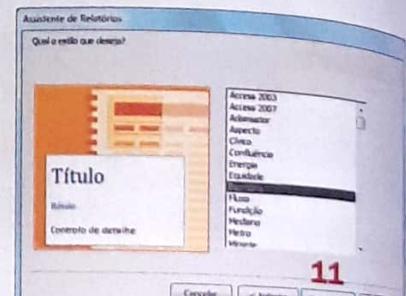


Fig. 98

5. Se no relatório estiver envolvida mais do que uma tabela, em **Como deseja ver os dados?**, deves seleccionar a opção pretendida.

Na figura 94, a opção seleccionada é **por Autores** e, por este motivo, o nome de cada autor aparecerá em destaque seguido das respectivas obras.

6. Clica em **Seguinte**.

7. Em **Deseja adicionar algum nível de agrupamento?**, selecciona as opções que pretendes.

No painel da direita analisa as alterações à estrutura do relatório.

8. Clica em **Seguinte**.

9. Define a sequência de ordenação que pretendes e, em seguida, clica em **Seguinte**.

10. Selecciona o tipo de esquema que pretendes para o relatório e, em seguida, clica em **Seguinte**.

11. Selecciona um estilo que julgues adequado e, em seguida, clica em **Seguinte**.

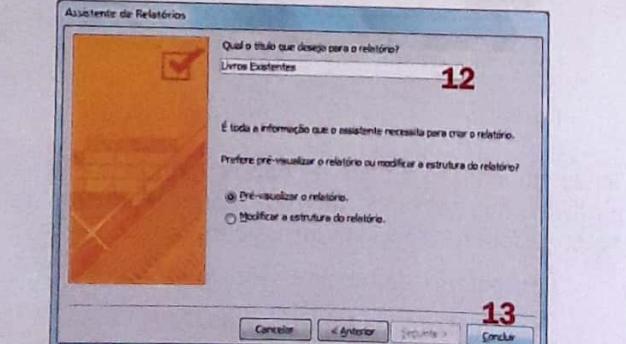


Fig. 99

Clica em para fechar a pré-visualização (Fig. 100).

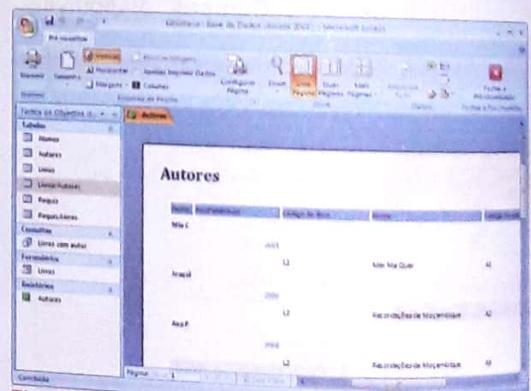


Fig. 100

Alguma informação do relatório pode não ficar totalmente visível. Contudo, como verás, podes aceder ao relatório em **Vista de Estrutura** e alterar o tamanho dos campos.

3.13.1. A estrutura de um relatório

Para aceder à estrutura de um relatório, selecciona, no separador Base, botão **Vistas**, a opção **Vista de Estrutura**.

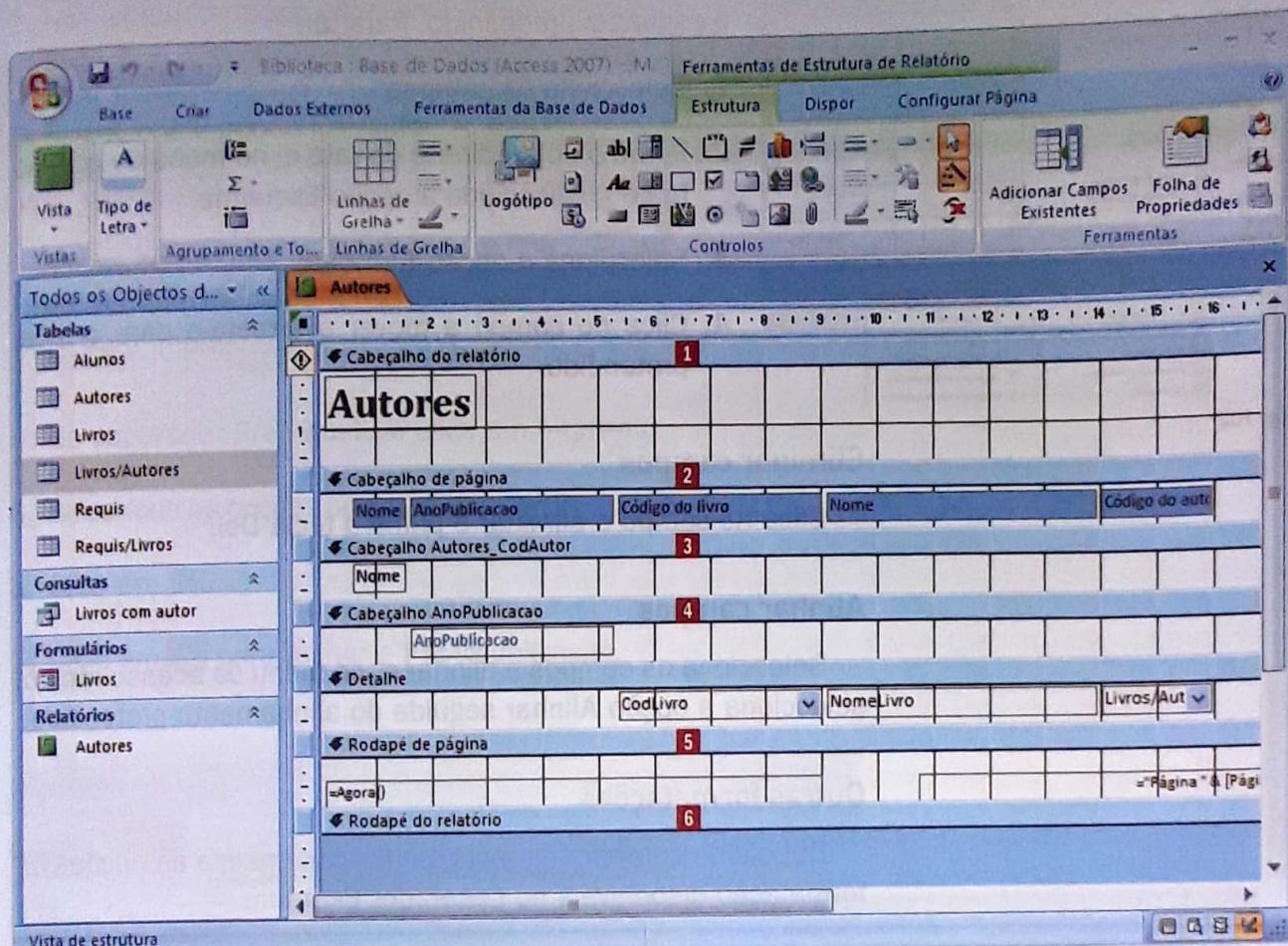


Fig. 101

1 Cabeçalho do relatório – secção não obrigatória que, caso exista, aparece apenas uma vez no início do relatório.

2 Cabeçalho de página – secção não obrigatória que, caso exista, aparece no início de cada página do relatório.

3 Cabeçalho descrição – aparece no início de um novo grupo de registos.

4 Detalhe – secção obrigatória que contém o corpo principal dos dados.

5 Rodapé de página – secção não obrigatória que, caso exista, aparece no fundo de cada página.

6 Rodapé do relatório – secção não obrigatória que, caso exista, aparece uma vez e no final do relatório.

Um relatório pode também ter rodapés de grupos que aparecem no final de cada grupo de registos.

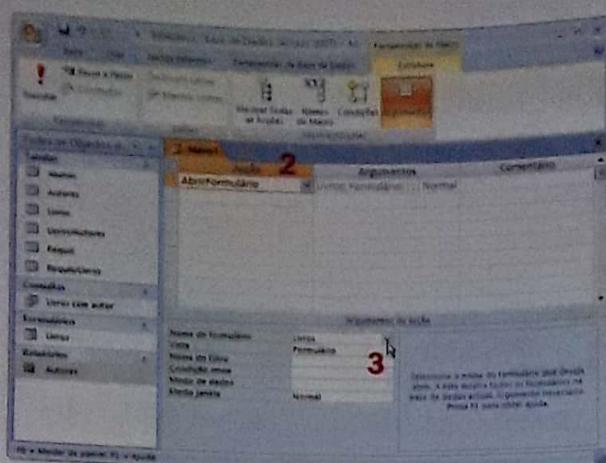


Fig. 107

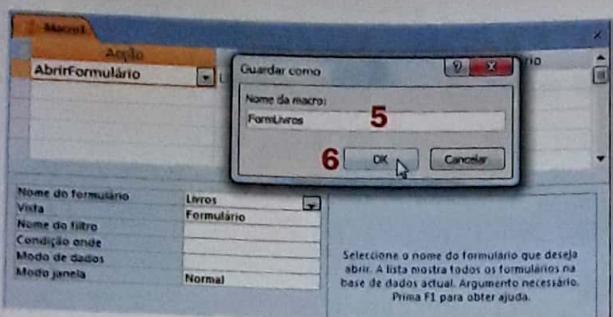


Fig. 108

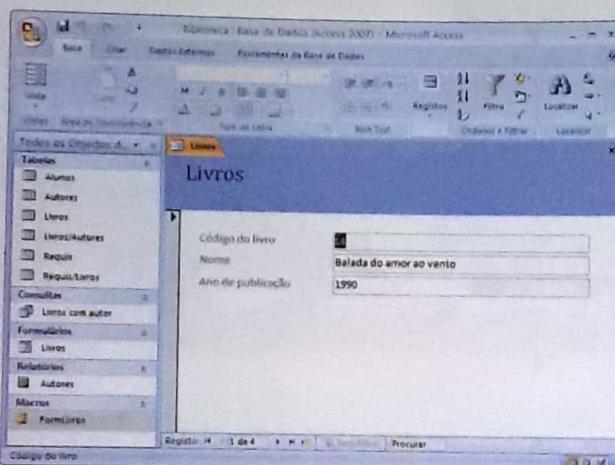


Fig. 109

Criação de macros

1. No separador Criar, grupo Outro, clica em Macros.
2. Na coluna Acção, na primeira linha disponível, clica na caixa de selecção e selecciona a opção pretendida.
3. Em Argumentos de acção, preenche os campos necessários.

Repete os passos 2 e 3 até especificares convenientemente todas as acções a serem executadas.

Na **macro** ilustrada na figura 107, a acção é **AbrirFormulário** e o nome do formulário é **Livros**. Assim, ao executar esta macro, o **formulário Livros** será aberto.

4. Clica em para guardar a macro.
5. Na janela **Guardar como**, digita o nome a atribuir à macro.
6. Clica em **OK**.

Executar uma macro

Para executar uma macro, basta fazer duplo clique no seu nome.

Criação de uma macro com condições

Quando se pretende que algumas funcionalidades de uma macro se executem mediante resultado de algumas condições, estas têm de ser especificadas.

Assim, quando uma macro está a ser executada, o Access avalia a primeira expressão:

- se for verdadeira, é executada a acção que se encontra nessa linha e todas as outras que se encontram nas linhas imediatamente a seguir e que sejam precedidas por reticências (...) na coluna **Condição**;
- se for falsa, a acção é ignorada, bem como todas as outras imediatamente a seguir que sejam precedidas por reticências (...) na coluna **Condição**.

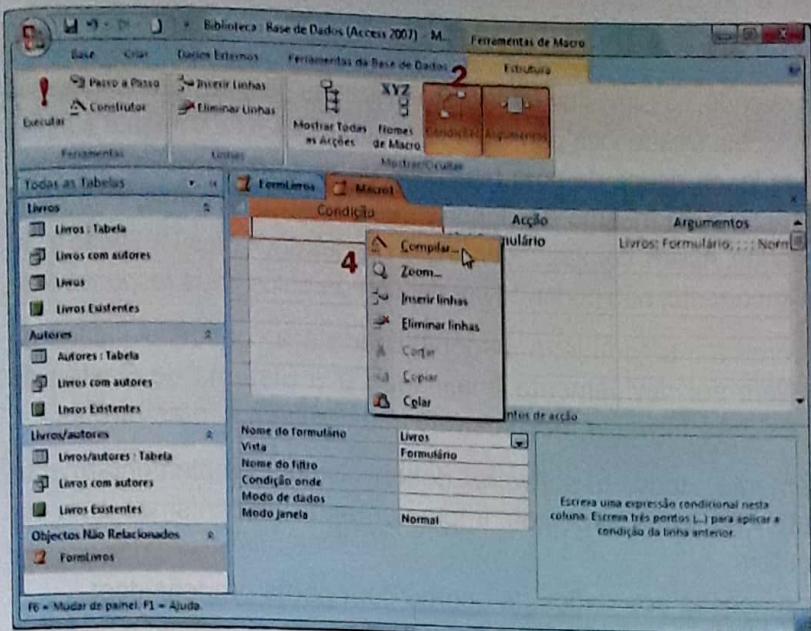


Fig. 110

1. Acede à macro em Vista de Estrutura.
2. No separador Estrutura clica em Condições.
3. Na coluna Acção indica a acção a ser executada.
4. Na coluna Condição, coloca o cursor na célula correspondente e prime o botão direito do rato. No menu de acesso rápido, selecciona a opção Compilar.
5. Cria a condição no Construtor de expressões.
6. Guarda a macro atribuindo-lhe um nome.

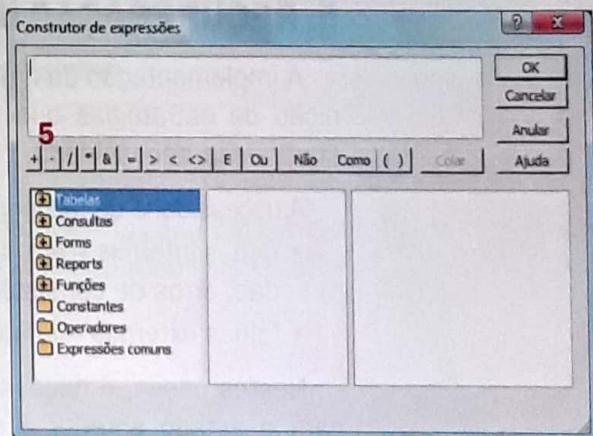


Fig. 111

3.15. Conceito de módulo

O recurso às macros permite automatizar muitas acções. Contudo, tratando-se de procedimentos mais específicos e/ou complexos, torna-se necessário recorrer a módulos de programação.

Como ainda não aprendestes a programar, fica apenas o apontamento de que os módulos utilizados em Access são programas escritos na linguagem de programação **Microsoft Visual Basic**.

4. SEGURANÇA DA INFORMAÇÃO

Como já foi referido, as bases de dados são utilizadas para armazenar diversos tipos de informações. Desde logo pensamos em situações relativamente simples como, por exemplo, os dados de uma biblioteca ou de uma papelaria. Contudo, também em bancos e até mesmo em departamentos governamentais são utilizadas bases de dados. Assim, compreenderás que a segurança da informação é um aspecto muito importante, não podendo ser descurada pelos seus responsáveis.

Questões como a integridade, a disponibilidade e a confidencialidade da informação devem ser devidamente analisadas e o Sistema de Gestão de Bases de Dados deve fornecer mecanismos que auxiliem nestas tarefas.

Simultaneamente, todo o software desenvolvido complementarmente deve ter em atenção estes aspectos, implementando mecanismos que, por exemplo, permitam:

- restringir o acesso aos dados de acordo com perfis predefinidos;
- conceder privilégios específicos aos utilizadores para acesso a objectos (tabelas, consultas, formulários, relatórios, etc.);
- impossibilitar os utilizadores da base de dados de alterar a sua estrutura.

5. RECUPERAÇÃO DA INFORMAÇÃO

A implementação de uma base de dados deve, desde logo, implicar a definição de estratégias que permitam recuperar a informação caso venha a surgir essa necessidade.

A necessidade de recuperação da informação pode ser devida a:

- **falhas internas** – erros de programação devido a situações não ponderadas, erros de consistência da informação, etc.;
- **falhas externas** – falha de energia, incêndio, catástrofe ambiental, etc.

Nestes casos, é necessário recuperar a informação, repondo a situação para o estado anterior à situação de erro detectada. Normalmente, esta recuperação é feita através de cópias de segurança da informação, realizadas previamente.

A **cópia de segurança** consiste na cópia dos dados de um dispositivo de armazenamento para outro. Os meios mais utilizados para a sua realização incluem CD-ROM, DVD, disco rígido, disco rígido externo, fitas magnéticas e a cópia de segurança externa (*online*). Neste caso, a informação é enviada, por uma rede como a Internet, para equipamentos mais sofisticados de grande porte e alta segurança.

A periodicidade de realização das cópias de segurança deve ser analisada em função da quantidade de dados alterados, mas o mais aconselhável é que seja diária e num horário em que não seja necessário aceder à informação a copiar. Normalmente, esta tarefa é realizada durante a noite, fora do horário laboral.

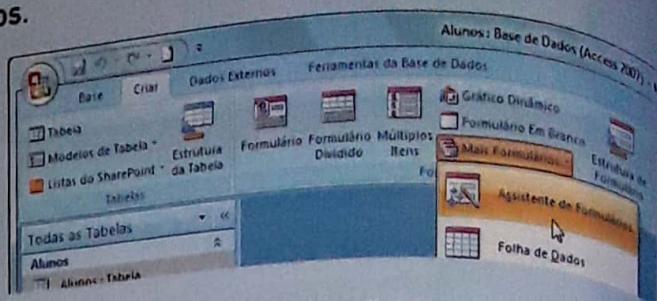
Para uma fácil localização da informação a repor, a cópia deve ser guardada, por data e categoria, em local seguro.

Actividades

8 Cria um formulário para a tabela Alunos.

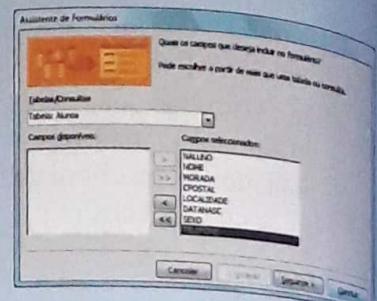
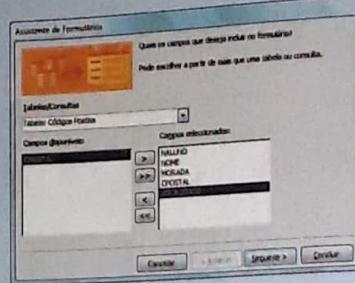
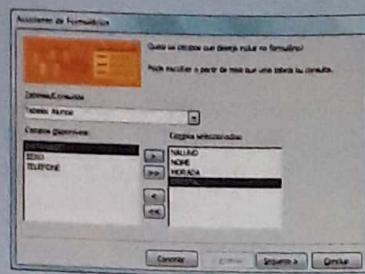
A localidade onde o aluno reside deve fazer parte do formulário.

- No separador Criar, grupo Formulários, clica em Mais Formulários.
- Selecciona a opção Assistente de Formulários.
- Em Tabelas/Consultas, selecciona a tabela Alunos.
- Selecciona os campos NALUNO, NOME, MORADA e CPOSTAL.



- Em Tabelas/Consultas, selecciona a tabela Códigos Postais.
- Selecciona o campo LOCALIDADE.

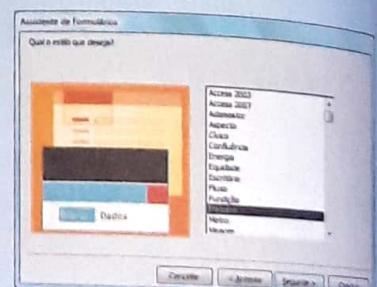
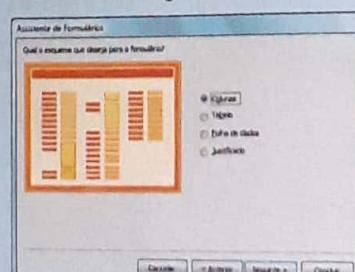
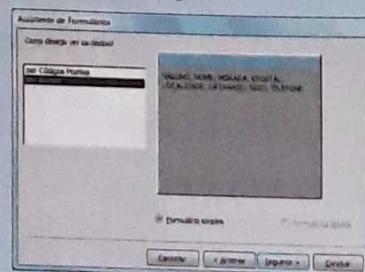
- Em Tabelas/Consultas, selecciona a tabela Alunos.
- Selecciona o campo DATANASC, SEXO e TELEFONE.
- Clica em Seguinte.



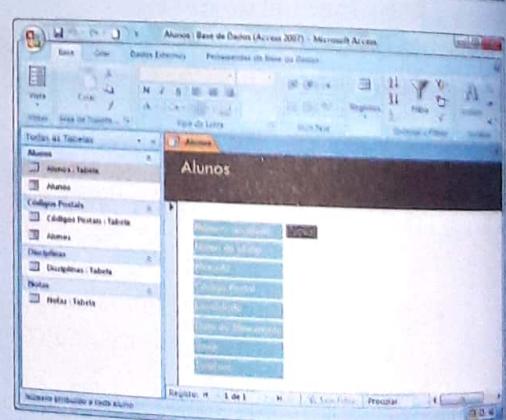
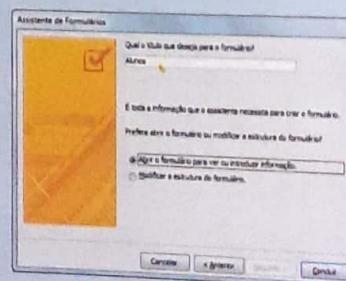
- Clica em Seguinte.

- Clica em Seguinte.

- Selecciona um estilo que te agrade.
- Clica em Seguinte.



- Clica em Concluir.



O resultado final será semelhante ao ilustrado na figura. Podes inserir informação na respectiva tabela a partir do formulário criado.

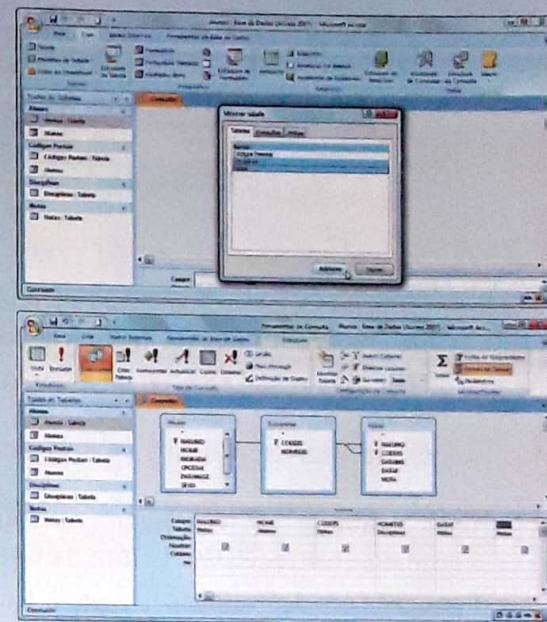
Os códigos postais inseridos na tabela Alunos devem existir na tabela Códigos Postais. Assim, **Vista de Folha de Dados** ou criar um formulário para o efeito.

9 Através do formulário ou acedendo às tabelas em **Vista de Folha de Dados**, insere alguns dados nas tabelas.

Tem em atenção que deves começar por inserir informação nas tabelas **Códigos Postais** e **Disciplinas**.

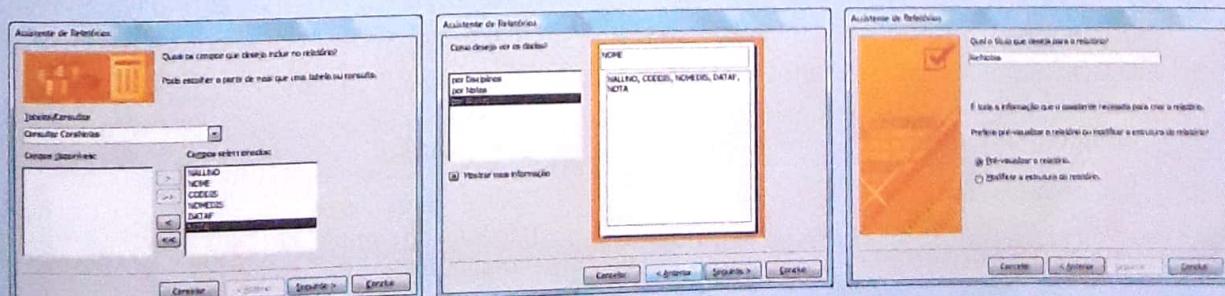
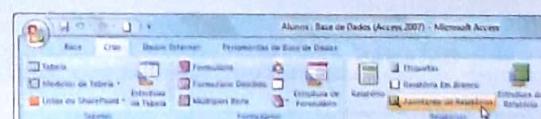
10 Cria uma consulta que permita visualizar o código do aluno e respectivo nome, o código da disciplina e respectiva descrição, a data de conclusão da disciplina e a nota obtida. Guarda a consulta com o nome **ConsNotas**.

- No separador **Criar**, grupo **Outro**, clica em **Estrutura da Consulta**.
- Na janela **Mostrar tabela** selecciona as tabelas **Alunos**, **Disciplinas** e **Notas**.
- Clica em **Adicionar** e, em seguida, em **Fechar**.
- Selecciona, da tabela **Notas**, o campo **NALUNO**.
- Selecciona, da tabela **Alunos**, o campo **NOME**.
- Selecciona, da tabela **Notas**, o campo **CODDIS**.
- Selecciona, da tabela **Disciplinas**, o campo **NOMEDIS**.
- Selecciona, da tabela **Notas**, os campos **DATAF** e **NOTA**.
- Guarda a consulta com o nome **ConsNotas** e executa-a para verificares que funciona correctamente.



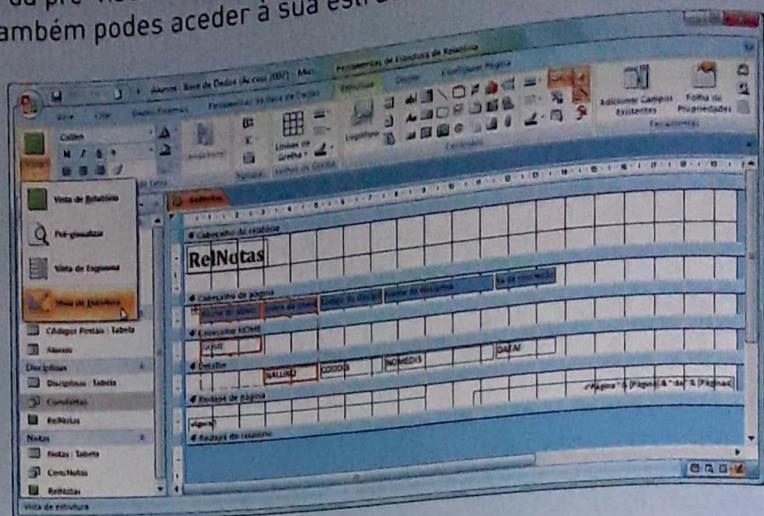
11 Cria um relatório baseado na consulta efectuada no ponto anterior. Guarda-o com o nome **RelNotas**.

- No separador **Criar**, grupo **Relatórios**, clica em **Assistente de Relatórios**.
- Selecciona a consulta **ConsNotas**.
- Selecciona todos os campos da consulta e clica em **Seguinte**.
- Selecciona o campo pelo qual pretendes visualizar a informação e clica em **Seguinte**.
- Nas janelas efectua as alterações que julgues adequadas e clica em **Seguinte**.
- Digita o nome a atribuir ao relatório e clica em **Concluir**.



Actividades

Fechando a janela da pré-visualização podes efectuar alterações à estrutura do relatório. Posteriormente, também podes aceder à sua estrutura através da Vista de Estrutura.



Actividade n.º 5

Uma instituição de apoio a pessoas necessitadas pretende manter actualizado o registo dos utentes (pessoas que procuram ajuda na instituição) e dos apoios (bens) que lhes são concedidos. Assume que os apoios são ao nível da alimentação e vestuário.

- 1 Acede ao Microsoft Access e cria uma base de dados chamada **APOIOAMIGO**.
- 2 Cria a tabela **UTENTES** com os seguintes campos: código do utente (**CODIGOUTENTE**), nome do utente (**NOME**), data de nascimento (**DATANAS**) e data de registo (**DATAREG**). Para cada um dos campos tem em atenção as seguintes especificações:

CODIGOUTENTE

Tipo de dados: Atribuição automática

NOME

Tipo de dados: Texto
Tamanho do campo: 60
Obrigatório: Sim

DATANAS

Tipo de dados: Data/Hora
Formato: Data abreviada
Obrigatório: Sim

DATAREG

Tipo de dados: Data/Hora
Formato: Data abreviada
Regra de validação: (Não superior à data actual)
Texto de validação: (A definir por ti)
Obrigatório: Sim

3

Cria a tabela **APOIOS** para registo dos apoios (bens) que a instituição tem disponíveis, com os seguintes campos: código do apoio (**CODIGOAPOIO**), designação do apoio (**DESIGNAÇÃO**), tipo de apoio – alimentar ou vestuário (**TIPOAPOIO**) e quantidade disponível (**QTDDISP**). Para cada um dos campos tem em atenção as seguintes especificações:

CODIGOAPOIO

Tipo de dados: Texto
Tamanho do campo: 5
Obrigatório: Sim

DESIGNAÇÃO

Tipo de dados: Texto
Tamanho do campo: 30
Obrigatório: Sim

TIPOAPOIO

Tipo de dados: Texto
 Tamanho do campo: 1
 Regra de validação: [Só devem ser aceites as letras A ou V]
 Texto de validação: Introduza A se o apoio for alimentar ou V se for vestuário.
 Obrigatório: Sim

QTDDISP

Tipo de dados: Número
 Tamanho do campo: Simples
 Formatar: Número geral
 Casas decimais: 2
 Obrigatório: Sim

- 4 Cria a tabela **APOIOS/UTENTES**, para registo dos apoios prestados aos utentes, com os seguintes campos: código do utente (CODIGOUTENTE), data em que o apoio foi prestado (DATA), código do apoio (CODIGOAPOIO) e quantidade (QUANTIDADE). Especifica, adequadamente, as propriedades de cada campo.

- 5 Define a(s) chave(s) primária(s) de cada uma das tabelas.

- 6 Estabelece as devidas relações entre as tabelas.

- 7 Cria formulários para as tabelas **UTENTES** e **APOIOS** a teu gosto.

- 8 Cria um formulário, com subformulário, para a tabela **UTENTES** e **APOIOS/UTENTES**, idêntico ao seguinte:

- 9 A partir dos formulários criados, insere alguns dados nas tabelas.

- 10 Cria uma consulta que permita visualizar os utentes inscritos. A informação deve estar ordenada por data de nascimento e nome.

- 11 Cria uma consulta que permita visualizar os utentes que tiveram apoios alimentares com indicação dos respectivos bens e quantidades.

- 12 Cria um relatório baseado na consulta efectuada na alínea anterior.

Formata o relatório a teu gosto tendo em atenção o seguinte:

- deverá ter um cabeçalho com um título adequado;
- no rodapé deverão ser impressas a data e a hora do sistema.

- 13 Desenvolve outras funcionalidades que julgues adequadas como, por exemplo, relatórios de utentes, de todos os apoios que usufruíram, etc.

- 14 Cria um formulário de botões que permita aceder aos formulários, consultas e relatórios criados.

Apóios a utentes

Código do utente:	<input type="text"/>																					
Nome:	<input type="text"/>																					
Apóios/utentes																						
Data	Código do apoio	Quantidade																				
<table border="1"> <tr> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> <td>6</td> <td>7</td> <td>8</td> <td>9</td> <td>10</td> <td>11</td> <td>12</td> <td>13</td> <td>14</td> <td>15</td> <td>16</td> <td>17</td> <td>18</td> <td>19</td> <td>20</td> </tr> </table>			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20			
Registo: 1 de 1 Anterior Próximo																						

216471
DE9053F
BAA3E 2-7
1376C7



2

INTRODUÇÃO A PROGRAMAÇÃO E SEGURANÇA DE COMPUTADORES



INTRODUÇÃO A PROGRAMAÇÃO

1. CONCEITOS BÁSICOS

1.1. Sistema informático

Inúmeros aspectos da sociedade moderna não seriam possíveis sem os computadores.

Nestes equipamentos, utilizados para processar informações, os dados são introduzidos através dos dispositivos de entrada (**entrada de dados**) e enviados para o processador. Após o seu **processamento**, a informação é emitida (**saída de informação**) numa forma que, supostamente, deve ser útil.

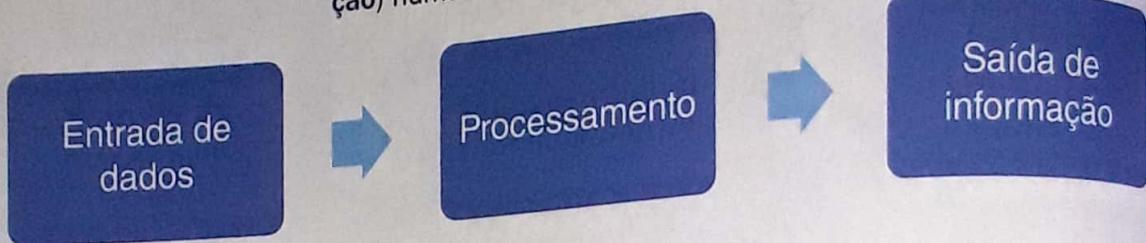


Fig. 1

Um **sistema informático** é constituído por um conjunto de componentes físicos (dispositivos mecânicos, magnéticos, eléctricos ou electrónicos), a que, genericamente, chamamos **hardware**, e por programas, métodos e procedimentos, regras e documentação, necessários ao funcionamento do **hardware**, a que, genericamente, chamamos **software** e cujo objectivo é oferecer um meio produtivo e propício para a computação.

Assim, **hardware** e **software** são, naturalmente, conceitos inseparáveis.

1.2. Ambiente de programação

Os computadores, em geral, falam uma linguagem eléctrica e os utilizadores, normalmente, falam a sua língua nativa. Assim, para que os computadores sejam a solução dos problemas, é necessário que se estabeleça uma comunicação entre eles e os seus utilizadores, e isto é feito através de um processo chamado **programação**.

Ao longo dos anos, têm sido criadas várias linguagens de programação, tendo algumas caído em desuso. Umas são relativamente próximas da linguagem eléctrica da máquina, outras, conhecidas por **linguagens de alto nível**, são mais próximas do inglês.

Uma **linguagem de programação** é um método, baseado em regras sintáticas e semânticas, para comunicar instruções para um computador.

A selecção da linguagem de programação a utilizar depende do tipo de programa a desenvolver.

As denominadas linguagens de programação de quarta geração (4GL) são linguagens de alto nível, com objectivos específicos, como o desenvolvimento de programas (softwares) comerciais.

Recorrendo a uma linguagem de programação (normalmente linguagem de alto nível), o programador fornece instruções detalhadas ao computador, especificando, passo a passo, exactamente aquilo que pretende que seja feito (**programa-fonte**). O computador, usando programas desenvolvidos pelo fabricante (**compiladores**), previamente instalados, efectua a tradução dessas instruções em **linguagem máquina** e verifica se as regras de sintaxe da linguagem utilizada foram respeitadas. Como resultado da **compilação** obtém-se o **programa-objecto**, que poderá, então, ser **executado**.

Os passos da criação e execução de um programa de computador podem ser representados pela figura seguinte:

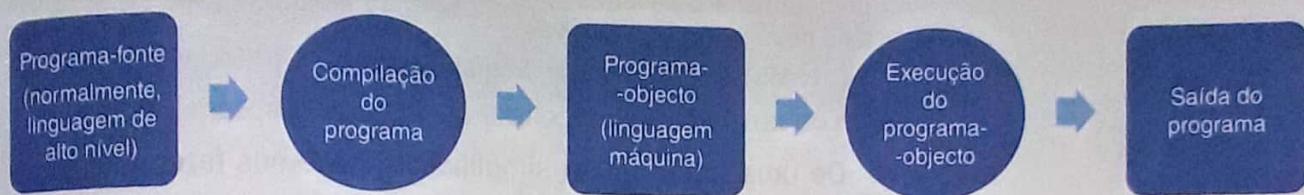


Fig. 2

A determinação do ambiente de programação mais adequado depende do tipo de aplicação a desenvolver, da linguagem de programação que o utilizador ou a organização preferem e das ferramentas disponíveis.

Algumas das linguagens de programação mais utilizadas são: C, Java, Objective-C, C++, C# (C Sharp), PHP, VisualBasic, Python e Perl.

2. ALGORITMOS

2.1. Conceito

O computador faz apenas aquilo que o(s) programa(s) em execução determina(m) que faça. Assim, é importante que não haja nenhum erro nem ambiguidade nas instruções dos programas que executa.

Dependendo da complexidade inerente ao problema em si, a criação de um programa informático pode parecer-nos muito complexa. Contudo, se dividirmos o problema, sistematicamente, em partes menos complexas (abordagem **dividir-para-conquistar**), o trabalho de programação poderá ser bastante facilitado.

Assim, numa fase inicial, é importante, senão essencial, a elaboração de um **algoritmo** que resolva o problema colocado, numa linguagem que nos seja fácil perceber (**linguagem natural**). Posteriormente, quando convictos que o algoritmo criado é o mais adequado, é que se deve passar à sua implementação numa linguagem de programação (**linguagem formal**).

Um **algoritmo** pode ser definido como uma sequência finita e ordenada, sem ambiguidades, de passos que levam à solução de um dado problema.

Não pensando em termos informáticos, podes considerar como algoritmo, por exemplo, a receita de um bolo ou as instruções para montar um brinquedo.

A **linguagem natural** foi desenvolvida pelo ser humano como instrumento de comunicação, como as línguas faladas e a língua de sinais. Ex.: português, inglês, etc.

A **linguagem formal** é uma linguagem definida a partir de um conjunto predeterminado de letras e/ou símbolos, onde apenas algumas sequências destes são significativas.

Ex.: todas as linguagens de programação.

De uma forma muito simplificada, podemos fazer a seguinte representação esquemática:

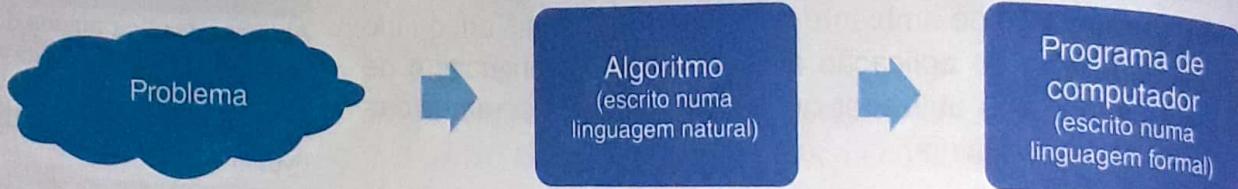


Fig. 3

Depois do algoritmo criado, a sua codificação, em qualquer linguagem de programação, é, praticamente, directa.

2.2. Construção de um algoritmo

Para construir um algoritmo que possa cumprir o seu propósito, é necessário ter em atenção algumas etapas importantes. Nomeadamente:

- compreender o problema;
- identificar os dados de entrada;
- identificar os dados de saída;
- determinar o que é preciso para transformar dados de entrada em dados de saída:
 - usar a estratégia do dividir-para-conquistar;
 - observar regras e limitações;
 - identificar todas as acções a realizar;
 - eliminar ambiguidades.

Os passos de um **algoritmo** devem ser definidos numa ordem específica, ser finitos, simples e sem ambiguidades.

Como exemplo, analisa um possível algoritmo, relativo à substituição de uma lâmpada de tecto que está fundida. Para facilitar o raciocínio, partimos do princípio que esta tarefa deveria ser realizada por um robô.

1. Seleciona uma nova lâmpada para substituir a avariada:
 - a. Se a potência não for a adequada, repete o processo até encontrar uma adequada:
 - i. coloca de lado a lâmpada;
 - ii. seleciona uma nova lâmpada.
2. Posiciona a escada debaixo da lâmpada.
3. Repete enquanto não alcançares a lâmpada.
 - a. Sobe um degrau da escada.
4. Repete enquanto a lâmpada está presa ao casquilho.
 - a. Gira a lâmpada no sentido contrário ao dos ponteiros do relógio.
5. Coloca a nova lâmpada no casquilho.
6. Repete enquanto a lâmpada não está presa ao casquilho.
 - a. Gira a lâmpada no sentido dos ponteiros do relógio.
7. Repete enquanto não alcançares o chão.
 - a. Desce um degrau da escada.

Muitos passos poderiam ser acrescentados a este algoritmo. Efectivamente, um algoritmo pode ser detalhado quase que indefinidamente.

3. VARIÁVEIS

Em termos informáticos, uma **variável** é uma entidade que possui um **valor** e é reconhecida por um **nome**.



Por exemplo, na relação: $A^2 = B^2 + C^2$, A, B e C são **variáveis**.

Durante a execução de um programa, uma variável pode ter valores diferentes mas, num dado instante, a variável apenas pode conter um valor.

Existem algumas regras a que deves atender quando da escolha dos nomes das variáveis. Nomeadamente:

- não pode ser uma palavra reconhecida pela linguagem de programação (ex.: se o programa for em Pascal, a variável não poderá ter como nome Begin, End, Var, Program, etc.);

 Nomes válidos para variáveis

TOTAL largura
PESO Medida
VALOR1 altura_lado

 Nomes inválidos para variáveis

6TOTAL X+Y
A%2 Y&Z
Altura do lado var

Dependendo da linguagem de programação utilizada, as regras acima referidas podem variar ligeiramente.

4. TIPOS DE DADOS

Tipo de dados		Valores possíveis
Numérico	ShortInt	- 128 a 127
	Integer	- 32768 a 32767
	LongInt	- 2147483648 a 2147483647
	Byte	0 a 255
Lógico		True (Verdadeiro) ou False (Falso)
Carácter	Char	Qualquer dígito numérico, letra do alfabeto ou um símbolo especial
Cadeia de caracteres	String	Qualquer combinação de letras, números e símbolos
Reais	Real	2,9E ⁻³⁹ a 1,7E ³⁸
	Single	1,5E ⁻⁴⁵ a 3,4E ³⁸
	Double	5,0E ⁻³²⁴ a 1,7E ³⁰⁸

O tipo de dados representa o tipo de informação que uma variável pode armazenar e estabelece limites, mínimos e máximos, de valores suportados. Todas as linguagens de programação possuem, internamente, uma definição de tipos de dados. Regra geral, os tipos de dados apresentados na tabela são os mais comuns.

5. OPERADORES

Os operadores indicam como uma operação actua sobre dois ou mais elementos e estão divididos em quatro categorias:

Aritmética	Permitem efectuar as operações básicas da Matemática.
Relacional	Permitem a comparação entre expressões.
Lógica	Empregues em expressões de lógica com utilização dos valores TRUE (para verdadeiro) e FALSE (para falso).
Texto	Actuam sobre cadeias de caracteres alfanuméricos.

Categoría	Operação	Operador	Exemplo
Aritmética	Adição	+	$4 + 6,5$
	Subtração	-	$25,6 - 12$
	Multiplicação	*	$16 * 23$
	Divisão	/	$8 / 2$
	Divisão inteira	l	$11 \text{ l } 2$
	Resto da divisão	MOD	$12 \text{ MOD } 4$
Relacional	Potenciação	^	2^3
	Maior	>	$10 > 2$ (True)
	Menor	<	$10 < 2$ (False)
	Maior ou igual	\geq	$12 \geq 10$ (True)
	Menor ou igual	\leq	$10 \leq 12$ (True)
	Igual	=	$a + b = b + a$ (True)
Lógica	Diferente	\neq	$14 \neq 16$ (True)
	Conjunção	AND	$7 > 5 \text{ AND } 4 > 3$ (True)
	Disjunção inclusiva	OR	$7 > 5 \text{ OR } 4 > 3$ (True)
	Disjunção exclusiva	XOR	$7 > 5 \text{ XOR } 4 > 3$ (False)
Texto	Negação	NOT	$a = b \text{ AND NOT } C$
	Concatenação	&	"Born" & "Dia" = "Born Dia"

Relativamente aos operadores aritméticos, aplicam-se as seguintes regras de prioridade:

- 1.º operadores de potenciação (^);
 - 2.º operadores de divisão e multiplicação (/ e *);
 - 3.º a prioridade de operação estabelece-se da esquerda para a direita;
 - 4.º operações dentro de parênteses têm prioridade relativamente a quaisquer outras.

Assim, aplicando as regras de prioridade, o resultado da expressão $-2 * 10 + 2^3 / 4 - 5$, seria determinado da seguinte forma:

$$\begin{array}{ccccccccc}
 -2 & * & 10 & + & 2 & ^ & 3 & / & 4 & - & 5 \\
 \diagup & & \diagup & & \diagup & & \diagup & & \diagup & \\
 -2 & * & 10 & + & 8 & / & 4 & - & 5 \\
 \diagup & & \diagup & & \diagup & & \diagup & \\
 -20 & + & 2 & - & 5 & & 2 & & \\
 \diagup & & \diagup & & \diagup & & & \\
 -18 & - & 5 & & & & & \\
 \diagup & & \diagup & & & & & \\
 -23 & & & & & & & \\
 \end{array}$$

6. EXPRESSÕES

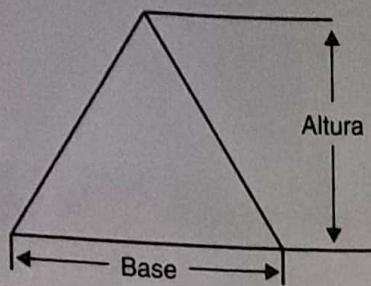


Fig. 4

Uma expressão resulta da combinação de constantes, variáveis, operadores, parênteses e nomes de funções especiais. O resultado de uma expressão é determinado considerando os valores das variáveis e constantes implicadas e a execução das operações indicadas.

$$(BASE * ALTURA) / 2$$

Esta é uma expressão bem tua conhecida e que permite determinar a área de um triângulo.

Nesta expressão estão envolvidas as variáveis **Base** e **Altura** e a constante 2. O resultado da expressão varia conforme os valores atribuídos às variáveis que a constituem.

7. OPERAÇÕES DE ATRIBUIÇÃO

A operação de atribuição especifica que a uma variável será atribuído um valor e é indicada pelo símbolo \leftarrow .

Exemplo:

BASE \leftarrow 12

Indica que à variável **BASE** é atribuído o valor **12**. Assim, após a execução deste comando, a variável **BASE** conterá o número **12**.



A **operação de atribuição** é uma operação **destrutiva** pois qualquer valor que a variável tenha, antes da sua execução, é substituído pelo novo valor.

A \leftarrow 10

A \leftarrow 0

A \leftarrow -1

No final, depois de executadas as três operações, a variável **A** conterá o valor -1.

Quando uma variável é do tipo **carácter**, o conteúdo atribuído deve ser indicado entre aspas ("").

NOME \leftarrow "Ana Maria"

DIASEMANA \leftarrow "Segunda-feira"

Para além de valores, a uma variável também pode ser atribuída uma expressão.

Exemplo:

BASE \leftarrow 4

ALTURA \leftarrow 6

AREA \leftarrow (BASE * ALTURA) / 2

À variável **AREA** vai ser atribuído o valor resultante da operação $(4 * 6) / 2$, ou seja, 12. As variáveis **BASE** e **ALTURA** mantêm os valores iniciais.

É importante realçar que os comandos são executados na sequência especificada.

```
BASE ← 4
AREA ← (BASE * ALTURA) / 2
ALTURA ← 6
```

Resultado do 1.º comando:
à variável **BASE** é atribuído o valor 4.

Resultado do 2.º comando:
como a variável **ALTURA** ainda não recebeu qualquer valor, não é reconhecida. Assim, apenas seria obtida uma indicação de erro.

7.1. Compatibilidade de tipos

Todas as variáveis devem ser classificadas com o tipo adequado e devem conter apenas valores do respectivo tipo.

Em termos informáticos, acções especiais são tomadas quando tentamos atribuir um valor a uma variável que tenha sido definida com um tipo diferente e, normalmente, resultará num erro de execução do programa.

Exemplo: Texto ← "Gosto do fim-de-semana"

```
Texto ← 7
```

Esta situação provocaria um erro, pois a variável **Texto**, definida como **string**, não suporta outros tipos de conteúdos.

Contudo, se atribuirmos um conteúdo real a uma variável inteira ou vice-versa, normalmente não há lugar a erro, mas o valor atribuído sofrerá uma conversão e será transformado, adaptando-se ao tipo da variável.

Exemplo: ValorInt ← -12 (constante inteira)

Uma **constante** é uma entidade reconhecida por um **nome** e que possui um **valor** que **não é alterado** durante a execução do programa.

Se a variável **ValorInt** tiver sido definida como **real**, de imediato o valor -12 é convertido para real, ajustando-se ao tipo da variável, e é guardado (armazenado) como -12,0.

```
ValorFim ← 234,65 (constante real)
```

Se a variável **ValorFim** tiver sido definida como **inteira**, de imediato o valor 234,65 é convertido para inteiro, truncando-se, ou seja, cortando o número no ponto decimal. Assim, apenas seria guardado (armazenado) o valor 234, que é inteiro.

 A conversão de números reais para inteiros e vice-versa pode alterar o valor original da variável. Esta situação deve ser tida em atenção, pois pode causar alterações ao valor final esperado.

Estas situações devem ser evitadas, a menos que sejam requeridas com um propósito especial.

Algoritmo MaiorDeDois: Dados dois números, determinar o maior.

```

Ler (N1, N2)
Se N1 > N2
  Então Escrever (N1)
  Senão Escrever (N2)
Fim
  
```

Cada um dos componentes desta estrutura pode ser mais complexo. Por exemplo, a condição pode ser composta e cada uma das alternativas pode ser constituída por vários comandos.

Traçagem do algoritmo utilizando os valores 10 e 12.

Passo	N1	N2	N1 > N2	Resultado
1	10	12		
2	10	12	[10 > 12] Falso	12

Nesta estrutura, a alternativa falsa pode não existir. Esta situação acontece quando não existe nenhuma acção caso a condição seja falsa. Em termos de fluxograma, este tipo de estrutura pode ser representado da seguinte forma:

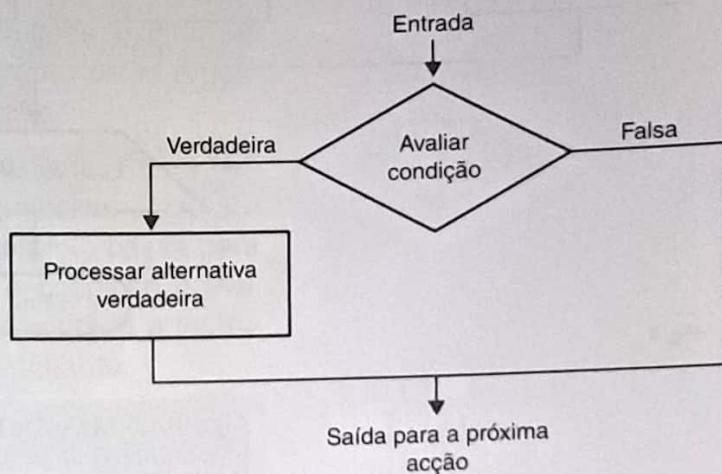


Fig. 8

10.2. Estrutura de decisão encadeada

Em situações mais complexas pode surgir a necessidade de utilizar uma estrutura **Se-Então-Senão**, encadeada numa alternativa original.

A sintaxe desta estrutura poderia ser:

```

Se Condição1
  Então Se Condição2
    Então AçãoV1
    Senão AçãoF1
  Senão Se Condição3
    Então AçãoV2
    Senão AçãoF2
  
```

Exemplo: Cria um algoritmo que determine o maior de três números. Os números devem ser introduzidos pelo utilizador e, no final, deve ser visualizado o maior.

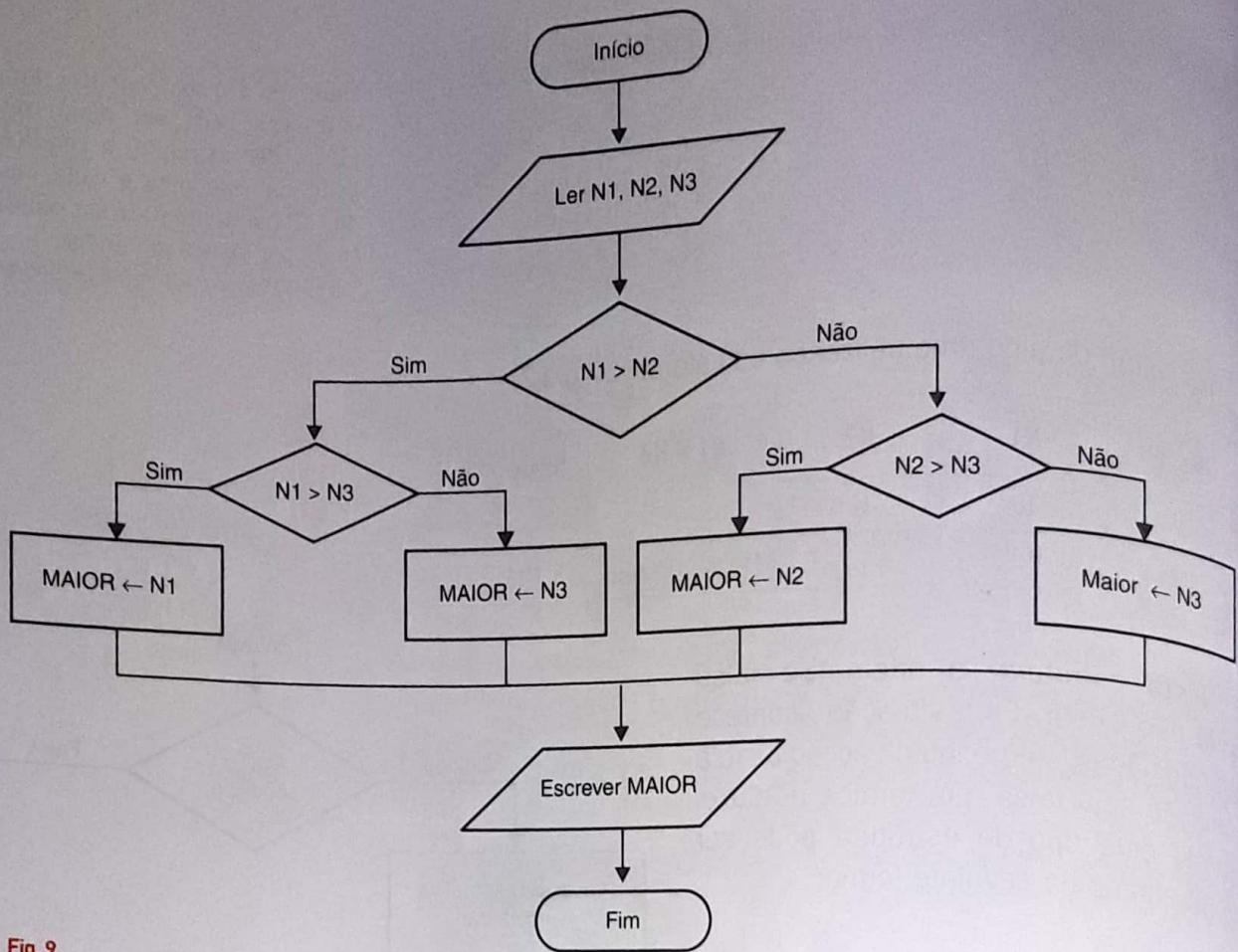


Fig. 9

Algoritmo MaiorDeTrês: Este algoritmo permite determinar o maior de três números lidos e imprimir o maior.

Assume-se que os três números são diferentes.

```

    Ler (N1, N2, N3)
    Se N1 > N2
    Então  Se N1 > N3
            Então MAIOR ← N1
            Senão MAIOR ← N3
    Senão  Se N2 > N3
            Então MAIOR ← N2
            Senão MAIOR ← N3
    Escrever ("O maior número é: ", MAIOR)
    Fim
  
```

Traçagem do algoritmo utilizando os valores 1, 2 e 3.

Passo	N1	N2	N3	N1 > N2	N1 > N3	N2 > N3	MAIOR	Resultado
1	1	2	3					
2	1	2	3	Falso				
3	1	2	3	Falso				
4	1	2	3	Falso		Falso		
5	1	2	3	Falso		Falso	3	
						Falso	3	0 maior número é: 3

11. ESTRUTURA DE REPETIÇÃO (CICLOS)

A estrutura de repetição é uma construção fundamental na programação, sendo utilizada sempre que uma ou uma série de operações devem ser repetidas diversas vezes.

11.1. Ciclos condicionais (Repita enquanto)

Sintaxe em notação algorítmica:

Repita enquanto condição

...

...

Fim enquanto

Utilizando esta estrutura, todas as instruções existentes entre a instrução **Repita enquanto** e **Fim enquanto** serão repetidas enquanto a **condição** expressa for verdadeira.

Antes de se iniciar o ciclo, a condição é avaliada. Se for verdadeira, as instruções são executadas sequencialmente, até ser encontrada a instrução **Fim enquanto**. O controlo passa para a instrução **Repita enquanto** e, novamente, a condição é avaliada. Logo que esta condição seja falsa, é executada a instrução imediatamente a seguir à instrução **Fim enquanto**.

Se logo na primeira verificação a condição retornar o valor falso, o ciclo não é executado e o programa prossegue com a execução da instrução imediatamente a seguir ao **Fim enquanto**.

O fluxograma geral pode ser representado da seguinte forma:

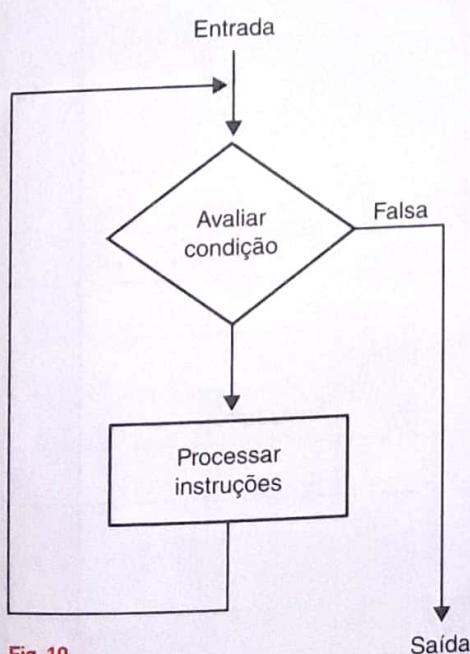


Fig. 10

Deve ser dada especial atenção à avaliação da condição e alteração dos conteúdos das variáveis de controlo. Uma situação não ponderada pode provocar uma repetição indefinida do ciclo (*do forever*).

Exemplo: Cria um algoritmo que, dadas duas sequências de três números, determine o maior de cada sequência.

Algoritmo MaiorDeTrês_2V: este algoritmo lê, por duas vezes, três números e imprime o maior de cada sequência. Assume-se que os três números são diferentes.

```

CONTADOR ← 0
Repete enquanto CONTADOR < 2
    Ler (N1, N2, N3)
    Se N1 > N2
        Então Se N1 > N3
            Então MAIOR ← N1
            Senão MAIOR ← N3
        Senão Se N2 > N3
            Então MAIOR ← N2
            Senão MAIOR ← N3
        Escrever ("O maior é: ", MAIOR)
    CONTADOR ← CONTADOR + 1
Fim enquanto
Fim

```

Traçagem do algoritmo utilizando os números 10, 14, 6 e 12, 8, 2.

Passo	CONTADOR	CONTADOR < 2	N1	N2	N3	N1 > N2	N1 > N3	N2 > N3	MAIOR	Resultado
1	0									
2		Verdadeiro								
3			10	14	6					
4							Falso			
5									Verdadeiro	
6										14
7										
8	1									O maior é: 14
9		Verdadeiro								
10			12	8	2					
11							Verdadeiro			
12								Verdadeiro		
13										12
14										
15	2									O maior é: 12
16		Falso								

A variável **CONTADOR** é uma variável de controlo, pois a sua função é controlar o número de vezes que o ciclo se repete. Inicialmente, é inicializada a 0 (zero), pois, caso contrário, não teria qualquer valor para poder ser testada e a execução da instrução **Repita enquanto** contador < 2 resultaria num erro.

No algoritmo **Maior de Três_2V** o número de vezes que o ciclo é repetido é conhecido, pois como a variável **CONTADOR** é inicializada a 0 (zero) e o processamento só será efectuado enquanto o seu conteúdo for menor que 2, só se repetirá duas vezes.

Contudo, o número de vezes que um ciclo se irá repetir pode depender da avaliação de uma ou mais condições que envolvam variáveis.

Exemplo:

Supõe que na disciplina de TIC os alunos de uma turma vão ser avaliados em função da nota de um teste e de um relatório, com pesos de 60% e 40%, respectivamente.

• Cria um algoritmo que leia o número de alunos (**NALUNOS**) da turma e, para cada um, leia o seu nome (**NOME**) e as notas obtidas no teste (**NOTAT**) e no relatório (**NOTAR**). Como resultado, para cada aluno deverá ser impresso o seu nome e a respectiva média ponderada.

Fluxograma:

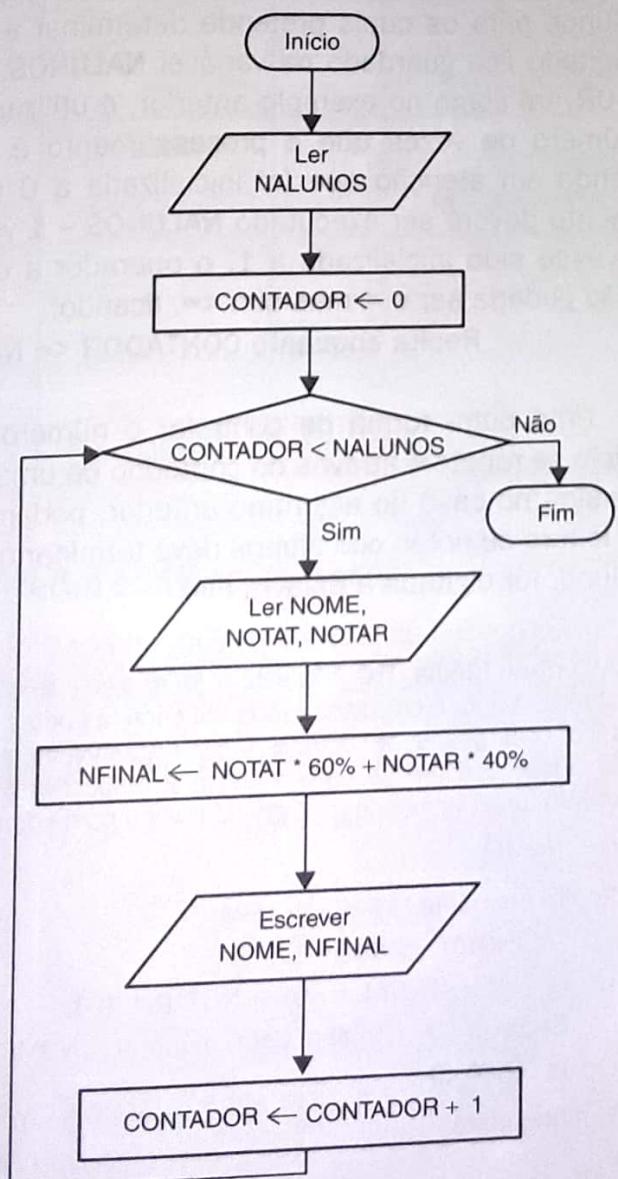


Fig. 11

Algoritmo Média_TIC_V1: este algoritmo calcula a nota final obtida pelos alunos. A nota é calculada tendo em conta as notas do teste e de um relatório com pesos de 60% e 40%, respectivamente. No final é impresso o nome de cada aluno e a respectiva nota final. É o utilizador que indica quantos alunos pretende processar.

```

Ler (NALUNOS)
CONTADOR ← 0
Repita enquanto CONTADOR < NALUNOS
    Ler (NOME, NOTAT, NOTAR)
    NFINAL ← NOTAT * 60% + NOTAR * 40%
    Escrever ("A nota final do(a) aluno(a)", NOME, "é", NFINAL)
    CONTADOR ← CONTADOR + 1
Fim enquanto
Fim

```

Inicialmente, é pedido ao utilizador que digite o número de alunos para os quais pretende determinar a nota final. O valor digitado fica guardado na variável **NALUNOS**. A variável **CONTADOR**, tal como no exemplo anterior, é utilizada para controlar o número de vezes que o processamento é efectuado. Assim, tendo em atenção que foi inicializada a 0 (zero), o processamento deverá ser executado **NALUNOS - 1** vezes. Se a variável tivesse sido inicializada a 1, o operador a utilizar na condição não poderia ser o < mas sim <=, ficando:

Repita enquanto CONTADOR <= NALUNOS

Uma outra forma de controlar o número de vezes que um ciclo se repete é através do conteúdo de uma variável de leitura. Assim, no caso do algoritmo anterior, podemos especificar que a leitura de notas dos alunos deve terminar quando, no nome do aluno, for digitada a palavra **FIM**.

Algoritmo Média_TIC_V2: este algoritmo calcula a nota final obtida pelos alunos. A nota é calculada tendo em conta as notas do teste e de um relatório com pesos de 60% e 40%, respectivamente. No final deve ser impresso o nome de cada aluno e a respectiva nota final. O processamento termina quando for digitado FIM no nome do aluno.

```

Ler (NOME)
Repita enquanto NOME <> "FIM"
    Ler (NOTAT, NOTAR)
    NFINAL ← NOTAT * 60% + NOTAR * 40%
    Escrever ("A nota final do(a) aluno(a)", NOME, "é", NFINAL)
    Ler (NOME)
Fim enquanto
Fim

```

O nome do aluno é lido inicialmente e a condição é verificada, se logo da primeira vez o utilizador digitar FIM, o ciclo não chega a ser executado. Se tal não acontecer, são lidas as notas, calculada e impressa a nota final. Em seguida, é pedido o nome do próximo aluno e, novamente, a condição é verificada.

Traçagem do algoritmo, apenas para dois alunos, considerando os seguintes dados: Ana, 12, 16; Miguel, 17, 18, FIM.

Passo	NOME	NOME <> "FIM"	NOTAT	NOTAR	NFINAL	Resultado
1	Ana					
2		Verdadeiro				
3			12	16		
4					13,6	
5						A nota final do(a) aluno(a) Ana é 13,6
6	Miguel					
7		Verdadeiro				
8			17	18		
9					17,4	
10						A nota final do(a) aluno(a) Miguel é 17,4
11	FIM					
12		Falso				

11.2. Ciclos contadores (Repita para)

Esta estrutura é utilizada quando, à partida, se sabe o número de vezes que o ciclo se vai repetir.

Sintaxe em notação algorítmica:

Repita para variável = numini, numini + 1, ..., numfim

...

...

Fim para

Exemplos:

Repita para NALUNO = 1, 2, ..., 25

Repita para VALOR = 1, 2, ..., N

Repita para CONTADOR = 2, 4, ..., 20

As variáveis **NALUNO**, **VALOR** e **CONTADOR** são variáveis de controlo de ciclo, pois apenas são necessárias para controlar o número de vezes que o ciclo se repetirá. Cada vez que o ciclo é executado, a variável de controlo assume, automaticamente, um novo valor.

Os números especificados no ciclo normalmente são inteiros, podem ser positivos ou negativos, em ordem crescente ou decrescente. O ciclo termina quando a lista de números tiver sido completada.

Em termos de programa, o tratamento deste tipo de ciclo é feito de diferentes formas, dependendo da linguagem de programação utilizada. Na maior parte das linguagens, é especificado um valor inicial, um incremento e um valor final, sendo cada novo valor calculado adicionando-lhe o incremento (que pode ser negativo). O ciclo termina quando o valor final é atingido ou ultrapassado. Quando o incremento é de 1, normalmente omite-se na instrução.

Exemplo:

Repita para **NUM** = 1 até 10 incremento 2

...

...

Fim para

O ciclo vai ser executado cinco vezes, assumindo a variável **NUM** os valores 1, 3, 5, 7 e 9, em cada ciclo, respectivamente.

Algoritmo Média_TIC_V3: este algoritmo calcula a nota final obtida pelos alunos. A nota é calculada tendo em conta as notas do teste e de um relatório com pesos de 60% e 40%, respectivamente. No final deve ser impresso o nome de cada aluno e a respectiva nota final. É o utilizador que indica quantos alunos pretende processar.

Ler (NALUNOS)

Repita para **NUM** = 1 até NALUNOS

Ler (NOME, NOTAT, NOTAR)

$NFINAL \leftarrow NOTAT * 60\% + NOTAR * 40\%$

Escrever ("A nota final do(a) aluno(a)", NOME, "é", NFINAL)

Fim para

Fim

Repara que a variável **CONTADOR** deixou de ser necessária, mas a sua função passou a ser executada pelo próprio ciclo.

Tracagem do algoritmo, apenas para dois alunos, considerando os seguintes dados: Ana, 12, 16; Miguel, 17, 18.

Passo	NALUNOS	NUM	NUM <= NALUNOS	NOME	NOTAT	NOTAR	NFINAL	Resultado
1	2							
2		1						
3			Verdadeiro					
4				Ana	12	16		
5							13,6	A nota final do(a) aluno(a) Ana é 13,6
6		2						
7			Verdadeiro					
8				Miguel	17	18		
9							17,4	A nota final do(a) aluno(a) Miguel é 17,4
10								
11		3						
12			Falso					
13								

Num algoritmo podemos utilizar estruturas de decisão e de repetição com condições simples ou compostas. Também, embutida numa estrutura, podemos utilizar várias outras.

Algoritmo Média_Final: este algoritmo calcula a nota final dos alunos a uma disciplina com indicação se o aluno está aprovado ou reprovado. A nota final corresponde à média aritmética das notas de cinco testes que os alunos realizaram. No final é impresso o nome de cada aluno, a nota final e a indicação de aprovação (se a nota final for superior ou igual a 9,5) ou reprovação (se a nota final for inferior a 9,5). O programa termina quando o utilizador digitar FIM no nome do aluno.

Ler (NOME)

Repete enquanto NOME <> "FIM"

 MEDIA ← 0

 Repete para TESTE = 1 até 5

 Ler (NOTA)

 MEDIA ← MEDIA + NOTA

 Fim para

 MEDIA ← MEDIA / 5

 Se MEDIA >= 9,5

 Então Escrever (NOME, MEDIA, "Aprovado")

 Senão Escrever (NOME, MEDIA, "Reprovado")

 Ler (NOME)

 Repete enquanto

 Fim

12. VECTORES

12.1. Operações com vectores

Um vector é uma estrutura unidimensional utilizada para armazenamento de dados. É identificado por um nome e cada um dos seus elementos é referenciado por um índice.

O conteúdo de um vector pode ser do tipo inteiro, real, string, etc. Contudo, todos os elementos que o constituem devem ser do mesmo tipo.

A sintaxe que vamos utilizar é a seguinte:

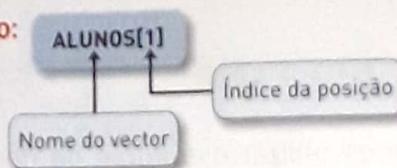
Nome_vector[Posição]

Nome_vector é o nome atribuído ao vector (escolhido pelo programador).

Posição é o número da posição no vector.

O armazenamento de dados num vector faz-se através de uma operação de atribuição, como se se tratasse de uma variável normal.

Exemplo:



X ← ALUNOS[1] é a instrução que atribui a variável **X** o conteúdo da primeira posição do vector **ALUNOS**.

Imagina que pretendes armazenar num vector o nome de 5 alunos e, num outro, as suas notas.

Teremos então os vectores **ALUNOS** e **NOTAS** cada um com 5 posições.

ALUNOS[1] ← ANA

NOTAS[1] ← 12

ALUNOS[2] ← ANDRÉ

NOTAS[2] ← 8

ALUNOS[3] ← MARIA

NOTAS[3] ← 16

ALUNOS[4] ← AUGUSTO

NOTAS[4] ← 20

ALUNOS[5] ← LUÍS

NOTAS[5] ← 14

Podes pensar nos vectores como estruturas semelhantes às seguintes:

ALUNOS	ALUNOS[1]	ALUNOS[2]	ALUNOS[3]	ALUNOS[4]	ALUNOS[5]
	Ana	André	Maria	Augusto	Luis
NOTAS	NOTAS[1]	NOTAS[2]	NOTAS[3]	NOTAS[4]	NOTAS[5]
	12	8	16	20	14

Algoritmo NOTA_MED_V1: este algoritmo determina a média das notas de cinco alunos a uma disciplina e imprime o nome e a nota dos alunos que se posicionem acima da média (nota **superior** à média). O nome (NOME) e a nota (N_ALUNO) de cada aluno devem ser introduzidos pelo utilizador.

SOMA ← 0

Repita para CICLO = 1 até 5

 Ler (NOME, N_ALUNO)

 ALUNO[CICLO] ← NOME

 NOTA[CICLO] ← N_ALUNO

 SOMA ← SOMA + N_ALUNO

Fim para

MEDIA ← SOMA / 5

Repita para CICLO = 1 até 5

 Se NOTA[CICLO] > MEDIA

 Então Escrever (NOME[CICLO], NOTA[CICLO])

Fim para

Fim

SOMA Variável que acumula o valor de cada nota lida

NOME Nome do aluno

N_ALUNO Nota do aluno

MEDIA Variável que guarda a média das notas

CICLO Variável que controla o número de vezes que o ciclo deve ser executado

Traçagem do algoritmo utilizando os valores anteriormente referidos.

Passo	SOMA	CICLO	NOME	N_ALUNO	MEDIA	NOTA[CICLO] > MEDIA	Resultado
1	0						
2		1					
3			Ana	12			
4	12						
5		2					
6			André	8			
7	20						
8		3					
9			Maria	16			
10	36						
11		4					
12			Augusto	20			

A medida que te sintas mais confiante na matéria, verás que os teus algoritmos irão adquirir uma estrutura mais simples e com as instruções estritamente necessárias.

12.2. Ordenação de vectores

A ordenação de um vector consiste em organizar os seus elementos utilizando um critério de ordenação. Por exemplo, por ordem alfabética ou numérica, crescente ou decrescente.

Existem diversos métodos de ordenação de vectores, com algoritmos desenvolvidos e testados, nomeadamente o método por selecção (Selection Sort), de que falaremos mais em pormenor, por inserção (Insertion Sort) e por bolha (Bubble Sort).

O método de ordenação de vectores por selecção é um dos métodos mais simples de ordenação e consiste em pesquisas sucessivas. Assim, inicia-se pela pesquisa do elemento de valor mais baixo que, em seguida, é permutado com o primeiro elemento. Dos restantes elementos, o de valor mais baixo é permutado com o segundo elemento e assim sucessivamente, até que o vector esteja ordenado.

Exemplo: Ordenar os elementos do vector LETRAS ascendente.

LETRAS[1]	LETRAS[2]	LETRAS[3]	LETRAS[4]	LETRAS[5]
D	A	E	B	C

1º passo

A partir da primeira posição do vector, procurar o menor elemento.

É encontrado o A na posição 2.

Trocar o conteúdo da posição onde foi encontrado o menor [2] com o conteúdo da posição 1.

LETRAS[1]	LETRAS[2]	LETRAS[3]	LETRAS[4]	LETRAS[5]
A	D	E	B	C

2º passo

A partir da segunda posição do vector, procurar o menor elemento.

É encontrado o B na posição 4.

Trocar o conteúdo da posição onde foi encontrado o menor [4] com o conteúdo da posição 2.

LETRAS[1]	LETRAS[2]	LETRAS[3]	LETRAS[4]	LETRAS[5]
A	B	E	D	C

3.º passo

A partir da terceira posição do vector, procurar o menor elemento.

É encontrado o C na posição 5.

Trocar o conteúdo da posição onde foi encontrado o menor [5] com o conteúdo da posição 3.

LETROS[1]	LETROS[2]	LETROS[3]	LETROS[4]	LETROS[5]
A	B	C	D	E

4.º passo

A partir da quarta posição do vector, procurar o menor elemento.

Não é encontrado nenhum elemento menor, logo não há alteração ao vector.

LETROS[1]	LETROS[2]	LETROS[3]	LETROS[4]	LETROS[5]
A	B	C	D	E

5.º passo

A partir da quinta posição do vector, procurar o menor elemento.

Como a quinta posição corresponde ao último elemento do vector não há qualquer processamento a efectuar. O vector encontra-se ordenado.

LETROS[1]	LETROS[2]	LETROS[3]	LETROS[4]	LETROS[5]
A	B	C	D	E

Algoritmo ORDENA_SELECCÃO: dado um vector K, contendo N elementos, este algoritmo ordena o vector por ordem ascendente.

Repete para pos1 = 1 até N - 1

 menor ← pos1

 Repete para pos2 = pos1 + 1 até N

 Se K[pos2] < K[menor]

 Então menor ← pos2

 Fim para

 temp ← K[pos1]

 K[pos1] ← K[menor]

 K[menor] ← temp

Repete para

 Fim

Neste algoritmo é utilizada uma variável temporária denominada `temp`. Esta variável é necessária para guardar o conteúdo do vector na posição `pos1` antes de este ser substituído pelo menor valor encontrado.

Tracagem do algoritmo utilizando o vector LETRAS com o conteúdo:

LETRAS[1]	LETRAS[2]	LETRAS[3]
D	A	C

$N = 3$

Passo	pos1	pos1 <= N - 1	menor	pos2	pos2 <= N	K[pos2] < K[menor]	temp	LETRAS[1]	LETRAS[2]	LETRAS[3]
1	1	Verdadeiro						D	A	C
2			1					D	A	C
3				2	Verdadeiro			D	A	C
4						A < D Verdadeiro		D	A	C
5			2					D	A	C
6				3	Verdadeiro			D	A	C
7						C < A Falso		D	A	C
8			4	Falso				D	A	C
9								D	A	C
10								A	D	C
11								A	D	C
12	2	Verdadeiro						A	D	C
13			2					A	D	C
14				3	Verdadeiro			A	D	C
15						C < D Verdadeiro		A	D	C
16			3					A	D	C
17				4	Falso			A	D	C
18								D	A	C
19								A	C	D

Relativamente à necessidade de utilizar variáveis temporárias, e para esclarecer alguma dúvida que ainda tenhas, analisa a seguinte situação:

Imagina que temos as variáveis A e B cujos valores são 10 e 20, respectivamente. Como proceder para que A passe a ter o conteúdo de B, ou seja, 20, e B passe a ter o conteúdo de A, ou seja, 10?

Façamos uma pequena traçagem:

$A \leftarrow 10$
 $B \leftarrow 20$
 $A \leftarrow B$
 $B \leftarrow A$



Passo	A	B
1	10	
2		20
3	20	
4		20

$A \leftarrow 10$
 $B \leftarrow 20$
 $temp \leftarrow A$
 $A \leftarrow B$
 $B \leftarrow temp$



Passo	A	B	temp
1	10		
2		20	
3			10
4	20		
5		10	

O conteúdo da variável A é destruído ao lhe ser atribuído o conteúdo de B. Contudo, ao guardar o conteúdo de A em temp, salvaguardamos o seu valor que, posteriormente, utilizamos para atribuir a B.

12.3. Pesquisa em vectores

A pesquisa num vector consiste em verificar se um dado valor existe, ou não, no vector e, caso exista, indicar em que posição do mesmo se encontra.

12.3.1. Pesquisa linear

Esta técnica consiste na verificação sequencial de cada elemento do vector. Considera-se que o vector não está ordenado.

Algoritmo PESQUISA_LINEAR: dado o vector K, com N elementos, este algoritmo procura, no vector, um elemento cujo valor é dado pela variável PESQ.

```

ENCONTRA ← 0
Repetir para CICLO = 1 até N
  Se K[CICLO] = PESQ
    Então Escrever ("Elemento encontrado na posição", CICLO)
    ENCONTRA ← 1
  Fim para
  Se ENCONTRA = 0
    Então Escrever ("O elemento não foi encontrado")
  Fim

```

A variável ENCONTRA é utilizada para determinar se o elemento foi ou não encontrado. Repara que se quando o ciclo terminar a variável continuar com o valor 0 (zero) é porque o teste $K[pos1] = PESQ$ nunca foi verdadeiro e, portanto, o seu valor não foi alterado.

2.3.2. Pesquisa binária

Esta técnica consiste na determinação do meio do vector comparando o conteúdo da posição encontrada com o elemento a pesquisar. Se o elemento a encontrar for menor, o procedimento é repetido na primeira metade até que o valor seja encontrado. Se o elemento a encontrar for maior, o procedimento é repetido na segunda metade até que o valor seja encontrado.

Esta técnica é mais eficiente do que a referida anteriormente. Contudo, só pode ser aplicada se o vector estiver ordenado.

Algoritmo PESQUISA_BINÁRIA: dado o vector K, com N elementos, este algoritmo procura, no vector, um elemento cujo valor é dado pela variável PESQ. As variáveis BAIXO, MEDIO e ALTO designam, respectivamente, o limite inferior, o valor médio e o limite superior do intervalo de pesquisa.

```

BAIXO ← 1
ALTO ← N
ENCONTRA ← 0
Repetir enquanto BAIXO <= ALTO
  MEDIO ← (BAIXO + ALTO) / 2
  Se PESQ < K[MEDIO]
    Então ALTO ← MEDIO - 1
  Senão Se PESQ > K[MEDIO]
    Então BAIXO ← MEDIO + 1
  Senão Escrever ("Elemento encontrado na posição", MEDIO)
  ENCONTRA ← 1
Fim enquanto
Se ENCONTRA = 0
Então Escrever ("O elemento não foi encontrado")
Fim

```

13. MATRIZES

Uma matriz é uma estrutura bidimensional utilizada para armazenamento de dados. É identificada por um nome e cada um dos seus elementos é referenciado por dois índices separados por uma vírgula. O primeiro índice especifica a linha e o segundo a coluna.

O conteúdo de uma matriz pode ser do tipo inteiro, real, string, etc. Contudo, os elementos de cada coluna devem ser do mesmo tipo.

A sintaxe que vamos utilizar é a seguinte:

Nome_matriz[Linha,Coluna]

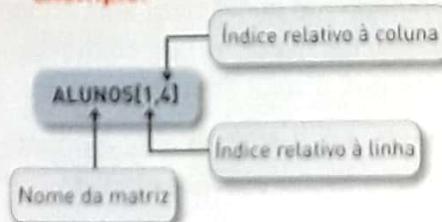
Nome_matriz é o nome atribuído à matriz (escolhido pelo programador).

Linha é o número que corresponde à linha.

Coluna é o número que corresponde à coluna.

O armazenamento de dados numa matriz faz-se através de uma operação de atribuição, como se se tratasse de uma variável normal.

Exemplo:



X ← ALUNOS[1,4] é a instrução que atribui à variável **X** o conteúdo armazenado na quarta coluna da primeira linha da matriz **ALUNOS**.

Imagina que pretendes armazenar numa matriz o nome de cinco alunos e as respectivas notas.

Teremos, então, a matriz **ALUNOS** constituída por cinco linhas (número de alunos) e duas colunas (a primeira para armazenar o nome e a segunda para armazenar a nota).

ALUNOS[1,1] ← ANA

NOTAS[1,2] ← 12

ALUNOS[2,1] ← ANDRÉ

NOTAS[2,2] ← 8

ALUNOS[3,1] ← MARIA

NOTAS[3,2] ← 16

ALUNOS[4,1] ← AUGUSTO

NOTAS[4,2] ← 20

ALUNOS[5,1] ← LUÍS

NOTAS[5,2] ← 14

Podes pensar na matriz como uma estrutura semelhante à que se apresenta ao lado.

ALUNOS	Coluna 1	Coluna 2
Linha 1	Ana	12
Linha 2	André	8
Linha 3	Maria	16
Linha 4	Augusto	20
Linha 5	Luis	14

Algoritmo NOTA_MED_V3: este algoritmo determina a média das notas de cinco alunos e imprime o nome e a nota dos alunos que se posicionem acima da média (nota **superior** à média). O nome (NOME) e a nota (N_ALUNO) de cada aluno devem ser introduzidos pelo utilizador.

```

SOMA ← 0
Repetir para CICLO = 1 até 5
  Ler [NOME, N_ALUNO]
  ALUNO[CICLO,1] ← NOME
  ALUNO[CICLO,2] ← N_ALUNO
  SOMA ← SOMA + N_ALUNO
Fim para
MÉDIA ← SOMA / 5
Repetir para CICLO = 1 até 5
  Se ALUNO[CICLO,2] > MÉDIA
    Então Escrever (ALUNO[CICLO,1], ALUNO[CICLO,2])
Fim para
Fim
  
```

A entrada ou o acesso à informação de uma matriz podem ser feitos elemento a elemento, utilizando ciclos.

Exemplo: Inserir informação na matriz **MATRIZ** de **M** linhas e **N** colunas. **M** e **N** são especificados pelo utilizador.

```

Ler (M,N)
Repetir para linha = 1 até M
  Repetir para coluna = 1 até N
    Ler (VALOR)
    MATRIZ[linha,coluna] ← VALOR
  Fim para
Fim para
  
```

Exemplo: Imprimir a informação contida na matriz **MATRIZ** de **M** linhas e **N** colunas.

```

Repetir para linha = 1 até M
  Repetir para coluna = 1 até N
    Escrever (MATRIZ[linha,coluna])
  Fim para
Fim para
  
```

14. SUB-ROTINAS

Sub-rotinas são conjuntos ou blocos de instruções que, normalmente, resolvem uma questão específica e que integram o nosso algoritmo ou programa.

A vantagem na definição de sub-rotinas está no facto de as podermos invocar (chamar) em diversas partes do algoritmo, não tendo, portanto, que repetir o seu código.

14.1. Funções

Para além das funções que, normalmente, todas as linguagens reconhecem (ver pág. 74), também o utilizador pode criar as suas próprias funções.

As funções são definidas como entidades separadas do algoritmo principal, identificadas por um nome, com parâmetros, fornecidos entre parênteses, imediatamente a seguir ao nome. Dado um conjunto de argumentos, uma função retorna um valor simples, especificado entre parênteses no comando **Retornar**.

Quando uma função é invocada (chamada), o controlo do algoritmo passa para a linha onde está especificada e todas as instruções que a constituem são executadas. Depois de executada, com os argumentos fornecidos, o controlo retorna, exactamente, ao ponto de chamada.

Exemplo:

FUNÇÃO MEDNOTAS(val1, val2, val3): esta função calcula a média dos três valores passados como parâmetros.

MEDIA \leftarrow (val1 + val2 + val3) / 3

Retornar (MEDIA)

Neste exemplo, **val1**, **val2** e **val3** são os parâmetros sobre os quais a função vai operar. O comando **Retornar** indica o ponto em que o controlo retorna ao algoritmo original, onde o valor retornado é utilizado.

Um algoritmo que utilize a função média poderia ser o seguinte:

Algoritmo NOTAS_V1: este algoritmo determina a média de três notas de 25 alunos e, no final, imprime o nome de cada aluno e a sua média. O nome e as notas de cada aluno devem ser introduzidos pelo utilizador. Repita para CICLO = 1 até 25

Ler (NOME, NOTA1, NOTA2, NOTA3)

FINAL \leftarrow MEDNOTAS (NOTA1, NOTA2, NOTA3)

Escrever ("Aluno(a) ", NOME, " - ", FINAL)

Fim para

Fim

14.2. Procedimentos

Os procedimentos são definidos como entidades separadas do algoritmo principal, identificados por um nome e, apesar de poderem ter parâmetros, normalmente tal não acontece.

O procedimento é chamado através do comando **Chame**, sendo o controlo do algoritmo transferido para a linha onde está especificado e todas as instruções que o constituem são executadas. Depois de executado, com os argumentos fornecidos, o controlo retorna ao ponto de chamada, sendo executada a instrução imediatamente a seguir ao comando **Chame**.

O procedimento não retorna nenhum valor. Qualquer valor a ser retornado por um procedimento volta através da lista de parâmetros.

Exemplo:

Algoritmo NOTAS_V2: este algoritmo determina a média de três notas de 25 alunos e, no final, imprime o nome de cada aluno e a sua média. O nome e as notas de cada aluno devem ser introduzidos pelo utilizador.

Repetir para CICLO = 1 até 25
Ler (NOME, NOTA1, NOTA2, NOTA3)

FINAL ← 0
Chame PMEDNOTAS_V2(NOTA1, NOTA2, NOTA3, FINAL)
Escrever ("Aluno(a)", NOME, " - ", FINAL)

Fim para

Fim

Procedimento PMEDNOTAS(val1, val2, val3, res): este procedimento calcula a média dos três valores passados como parâmetros. A média é atribuída à variável **res**.

res ← (val1 + val2 + val3) / 3

Retornar

Repara que:

- no algoritmo é passado o controlo ao **Procedimento PMEDNOTAS** com os parâmetros NOTA1, NOTA2, NOTA3 e FINAL;
- no **Procedimento PMEDNOTAS** é calculada a média das notas e o resultado atribuído à variável **res**;
- quando o procedimento termina e o controlo volta ao algoritmo principal, a variável **FINAL** tem o mesmo conteúdo da variável **res**.

Efectivamente, é feita a correspondência entre as variáveis NOTA1 e val1, NOTA2 e val2, NOTA3 e val3 e FINAL e res.

A conversão de graus Fahrenheit em graus Celsius é dada pela fórmula:

$$^{\circ}\text{C} = \frac{(^{\circ}\text{F} - 32)}{1,8}$$

a. Cria um algoritmo em pseudocódigo que leia o valor da temperatura em graus Fahrenheit e o converta em graus Celsius.

b. Faz uma traçagem do algoritmo considerando como lido o número 68.

11 Cria um algoritmo em pseudocódigo que, dado um número N (entre 1 e 10), apresente a sua tabuada.

Ex.: Para N = 5 deverá ser impresso

$$1 \times 5 = 5, 2 \times 5 = 10, \dots, 10 \times 5 = 50$$

12 Cria um algoritmo em pseudocódigo que leia vários números inteiros e calcule a soma dos números negativos. O fim da leitura é indicado pelo número zero.

13 Cria um algoritmo em pseudocódigo que calcule o peso ideal de uma pessoa. A altura e o sexo devem ser lidos e o peso ideal calculado tendo em atenção as fórmulas:

$$\text{peso ideal do homem} = (72,7 \times \text{altura}) - 58$$

$$\text{peso ideal da mulher} = (62,1 \times \text{altura}) - 44,7$$

14 Cria um algoritmo em pseudocódigo para calcular a média de um aluno baseada em 3 notas. Dependendo do resultado da média, o aluno poderá ser aprovado, reprovado ou admitido a exame.

Se a média das notas for maior ou igual a 7 e menor que 10, o aluno deverá fazer o exame.

Se a média das notas for inferior a 7, o aluno está reprovado.

Se a média das notas for maior ou igual a 10, o aluno está aprovado.

No final deverá ser visualizada a seguinte informação: Nome do aluno, média e respectiva situação (aprovado, reprovado ou exame).

15 Cria um algoritmo em pseudocódigo que leia o nome e a idade dos alunos até que seja introduzida, como nome do aluno, a palavra "STOP". No final devem ser mostrados o nome e a idade do aluno mais velho.

16 Cria um algoritmo em pseudocódigo que crie um vector de n números reais e determine o maior e o menor valores que o constituem.

17 Cria um algoritmo em pseudocódigo que leia uma relação de nomes de alunos e imprima a mesma lista mas por ordem crescente de nome.

18 Cria um algoritmo em pseudocódigo que leia dois vectores de 10 posições e faça a multiplicação dos elementos de mesmo índice, colocando o resultado num terceiro vector. No final deve ser visualizado o vector resultante.

19 Cria um algoritmo em pseudocódigo que leia a matriz $M(5,5)$ e calcule as seguintes somas:

a. Da linha 4 de M

b. Da coluna 2 de M

c. Da diagonal principal

d. De todos os elementos da matriz M

No final imprime a matriz e os resultados das somas.

SEGURANÇA DE COMPUTADORES

15. SEGURANÇA E GESTÃO DE ANTIVÍRUS

15.1. Vírus informáticos

Os vírus informáticos são programas criados com o intuito de interferirem com o normal funcionamento dos computadores, danificando-os ou, pelo menos, causando alguns contratemplos.

Não abras anexos de correio electrónico provenientes de um remetente desconhecido, pois pode tratar-se de um vírus.

Um programa deste tipo, ou seja, um vírus, pode danificar ou eliminar dados do computador, afectando o seu funcionamento ou, através de programas de correio electrónico, propagar-se a outros computadores. Espalham-se facilmente através de anexos em mensagens de correio electrónico ou mensagens instantâneas, através de transferências na Internet, etc. Assim, os vírus difundem-se sob a forma de cartões electrónicos ou qualquer tipo de ficheiro (áudio, vídeo, texto, etc.), programas transferidos, etc.

15.1.1. Tipos de vírus informáticos

Cavalos de Tróia

São um tipo de praga digital que, basicamente, permitem o acesso remoto ao computador após a infecção. Os cavalos de Tróia podem ter outras funcionalidades, como captura de dados do utilizador e execução de instruções presentes em scripts. Este tipo de programas parece ser útil, mas, na realidade, comprometem a segurança e causam muitos danos. Apresentam-se, por exemplo, como um e-mail com anexos com supostas actualizações de software de segurança da Microsoft, mas na realidade são programas que pretendem desactivar antivírus e firewalls.

Worms

Um worm cria cópias de si mesmo de um computador para outro, automaticamente. Primeiro, ele controla recursos no computador que permitem o transporte de arquivos ou informações. O grande perigo dos worms é a sua capacidade de se multiplicarem em grande volume. Por exemplo, um worm pode enviar cópias de si mesmo a todas as pessoas que constam no seu catálogo de endereços de e-mail e os computadores dessas pessoas passam a fazer o mesmo, causando um efeito domínio de alto tráfego de rede que pode tornar mais lentas as redes corporativas e a Internet como um todo.



Fig. 12

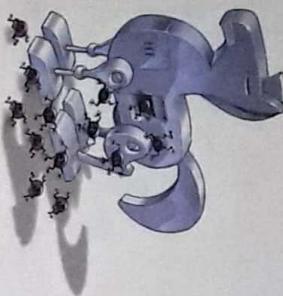


Fig. 12



Hijackers

Hijackers são programas ou scripts que “sequestram” navegadores de Internet, principalmente no Internet Explorer. Quando isso

Fig. 14

Por exemplo, o hijacker altera a página inicial do browser e impede o usuário de mudá-la, exibe propagandas em pop-ups ou janelas novas, instala barras de ferramentas no navegador e pode impedir o acesso a determinados sites (como sites de software antivírus, por exemplo).

Keyloggers

São pequenos aplicativos que podem ser embutidos em vírus, softwares ou softwares suspeitos, destinados a capturar tudo o que é digitado no teclado. O objectivo principal, neste caso, é capturar senhas.

Spywares

Consiste num programa automático de computador que recolhe informações sobre o usuário, sobre os seus costumes na Internet e transmite essa informação a uma entidade externa na Internet, sem o seu conhecimento nem consentimento.

15.1.2. Evitar vírus informáticos

Para evitar que o computador seja infectado por vírus, é essencial ter um programa antivírus instalado e devidamente actualizado. É também importante que respeites as regras básicas de navegação na Internet e que te mantenhas informado sobre as ameaças mais recentes.

Se ainda não instalaste uma ferramenta antivírus no teu computador, não adies mais. É agora o momento de o fazer! Faz o download de um dos antivírus referidos abaixo e instala-o seguindo as indicações. Em seguida, efectua uma análise ao computador.

Existem diversas ferramentas antivírus gratuitas, como, por exemplo, o **AVG**, o **Avira AntiVir** e o **Avast**. Contudo, se utilizas o Windows como sistema operativo, seria boa ideia conheceres o **Essentials**, antivírus aconselhado pela Microsoft. Como referência de ferramentas não gratuitas, temos, por exemplo, o **McAfee** e o **Norton**.

Relativamente a todas as ferramentas referidas, efectua uma pesquisa na Internet e acede aos respectivos sites onde encontrarás informação pormenorizada sobre cada uma.

Mesmo tendo um antivírus instalado no computador, quer por uma qualquer falha momentânea do sistema quer por incapacidade de detecção do programa instalado, um vírus pode “infectar” o computador. Nesta situação, é importante removê-lo de imediato para evitar mais infecções.



Fig. 15



Fig. 16

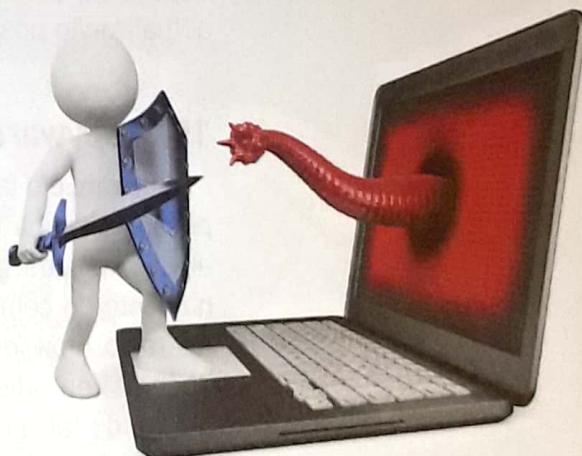


Fig. 17

Importante: Remove qualquer software antivírus que tenhas no computador antes de instalar um novo. Utilizar dois programas diferentes não tornará o sistema mais seguro. Bem pelo contrário, poderá provocar incompatibilidades.

15.1.3. Remover vírus informáticos

Remover um vírus informático de um computador pode não ser uma tarefa fácil, obrigando, quantas vezes, à formatação do disco e reinstalação do sistema operativo!

As ferramentas antivírus dão-nos uma grande segurança quanto à imunidade do computador. Contudo, diariamente são “inventados” novos vírus que a ferramenta instalada não reconhece e, portanto, não detecta.

As empresas que criam as ferramentas antivírus estão em alerta permanente e, a cada novo vírus detectado, é criada uma forma de o combater e o ficheiro de definições da ferramenta é actualizado para que o novo vírus passe a ser reconhecido. Daí que a actualização da ferramenta de antivírus seja essencial.

Mesmo tendo a actualização automática activada, quer do sistema operativo quer da ferramenta antivírus, deves, periodicamente:

- aceder ao Microsoft Update e instalar as actualizações recomendadas;
- aceder ao site do fabricante do antivírus instalado no computador e efectuar a sua actualização;
- executar uma análise aprofundada ao computador.

Se for detectado algum ficheiro com vírus, podes optar por eliminá-lo ou colocá-lo em quarentena, deixando-o guardado numa área específica, para que, eventualmente, possa ser recuperado numa actualização posterior do ficheiro de definições do antivírus.

15.2. Spyware

Spyware é o termo atribuído a programas que recolhem informações pessoais sem o nosso conhecimento ou permissão. Para além disto, alguns tipos de **spyware** podem efectuar alterações, tornando o funcionamento do computador mais lento ou até bloqueando-o e, também, alterar o *browser* instalado (por exemplo, modificando a *homepage* definida) ou adicionar componentes desnecessários (por exemplo, barras de ferramentas) ou mesmo indesejados.

O nosso computador pode tornar-se vítima de **spyware** ao ser feita a transferência de ficheiros de sites não fidedignos, como, por exemplo, ao fazer a transferência de jogos gratuitos ou músicas. Alguns sites com apresentações de publicidade (**adware**) também são uma fonte de **spyware**.

Contudo, não podemos partir do princípio de que todo o software que fornece anúncios ou detecta a nossa actividade *online* seja nocivo. Por exemplo, o comércio na Internet, desde que se concorde com as condições impostas, não tem de ser perigoso. Nesta situação, se concordarmos em permitir a detecção da nossa actividade *online*, a empresa pode, mais facilmente, determinar os anúncios que nos interessem.

Ter uma ferramenta antivírus instalada no computador não garante a imunidade. A sua actualização sistemática é OBRIGATÓRIA.

15.2.1. Evitar spyware

As formas de propagação de *spyware* são diversas e, normalmente, não temos percepção de que tal possa ter acontecido. Contudo, podemos tomar algumas medidas para minimizar os seus efeitos, nomeadamente:

Utilizar uma *firewall*

De uma forma simplificada, podemos dizer que a *firewall* examina a informação trocada entre o computador e a Internet, identificando e bloqueando informações que são fornecidas a partir de locais perigosos ou suspeitos. Para tal, a *firewall* deve ser configurada de forma que os piratas informáticos, na sua pesquisa de computadores vulneráveis, não consigam detectar o nosso computador.

Existem três tipos básicos de *firewalls*:

- *firewalls* de software;
- *routers* de hardware;
- *routers* sem fios.

A sua escolha deve ter em atenção o sistema que pretendemos proteger. O Windows Vista e Windows XP Service Pack 2 (SP2) têm uma *firewall* incorporada e activada por predefinição.

Actualizar o sistema

Certifica-te de que o software está sempre actualizado. Para tal, acede ao site da **Microsoft Update** e transfere todas as actualizações de segurança críticas disponíveis.

Ajusta as definições de segurança do Web browser utilizado

Normalmente, estas ferramentas dispõem, no seu menu, da opção **Ferramentas** ou **Configurações** onde podes definir algumas configurações relacionadas com **Segurança**.

Transferir e instalar uma ferramenta *antispyware*

Alguns fornecedores de acesso à Internet (ISP) incluem nos seus pacotes de serviços software *antispyware*. Neste caso, o mais aconselhável é saber, junto do respectivo ISP, se fornece este serviço e, caso isto não se verifique, saber qual o software que aconselha.

Existem diversos programas gratuitos como, por exemplo, Lavasoft, Ad-Aware e Spybot Search & Destroy, e não gratuitos como, por exemplo, McAfee e Norton, que permitem a remoção deste software indesejável. Contudo, se utilizas o Windows como sistema operativo, seria boa ideia conheceres o **Windows Defender**, *antispyware* aconselhado pela Microsoft.



3

TRABALHO DE PROJECTO

- evolução do grupo de trabalho, alterações de comportamento, aprendizagem realizada e a forma como foram geridos e resolvidos os momentos de tensão e conflito;
- cumprimento dos objectivos inicialmente previstos;
- forma como os recursos disponíveis foram utilizados;
- relatórios e capacidade de síntese na apresentação;
- postura e capacidade de comunicação.

Para além da avaliação dos resultados obtidos no projecto, o professor deve ainda considerar o envolvimento e participação de cada aluno. Desta forma tem condições para avaliar individualmente o contributo de cada aluno e, se necessário, atribuir uma avaliação diferente para cada um, dentro do mesmo grupo de trabalho.

O professor deve ainda promover a auto-avaliação dos alunos, permitindo que estes façam uma reflexão sobre o seu contributo e a aprendizagem realizada durante a execução do projecto.

O processo de avaliação é ainda ideal para os envolvidos no projecto reflectirem sobre os erros e falhas cometidos e a partir destes ponderarem sobre a forma como podem e devem ser evitados em situações futuras.

3. PAPEL DOS ALUNOS

No trabalho de projecto, todos os elementos do grupo devem participar activamente, sendo incentivados a expressar livremente as suas ideias e opiniões.

O trabalho dos alunos desenvolve-se em pequenos grupos, em que os elementos que os constituem se apoiam e cooperam. Os alunos colaboram e, juntos, procuram desenvolver o projecto que se propuseram concretizar. É, portanto, uma aprendizagem cooperativa, isto é, o conhecimento constrói-se no processo de interacção entre os alunos, entre estes e o professor, bem como com outros elementos da comunidade. Um dos aspectos mais marcantes do Trabalho de Projecto é o facto de se fundar no trabalho de grupo, o que permite desenvolver o sentido de responsabilidade, a solidariedade e o espírito de equipa.

Este papel activo dos alunos confere-lhes mais responsabilidades: efectivamente, a autonomia do trabalho tem como complemento a responsabilização. Por outro lado, e este é um dos aspetos mais importantes do Trabalho de Projecto, os conhecimentos, as experiências e os recursos dos alunos são valorizados, constituindo estímulos para a aquisição de novos conhecimentos.

Consulta o Apoio 1 da página 130:
Regras de trabalho em grupo

4. AS ETAPAS DE DESENVOLVIMENTO DE UM PROJECTO

Como já dissemos, um projecto visa identificar um problema, uma situação. A escolha e a realização do trabalho desenvolvem-se em grupo, implicando uma prévia negociação entre os seus membros. Assim, o projecto deve corresponder às necessidades e interesses do grupo, não podendo ser imposto, porque implica a adesão e participação activa, motivada e interessada de todos, promovendo o pensamento crítico e criativo.

Consideramos essenciais as seguintes etapas na elaboração de um projecto:

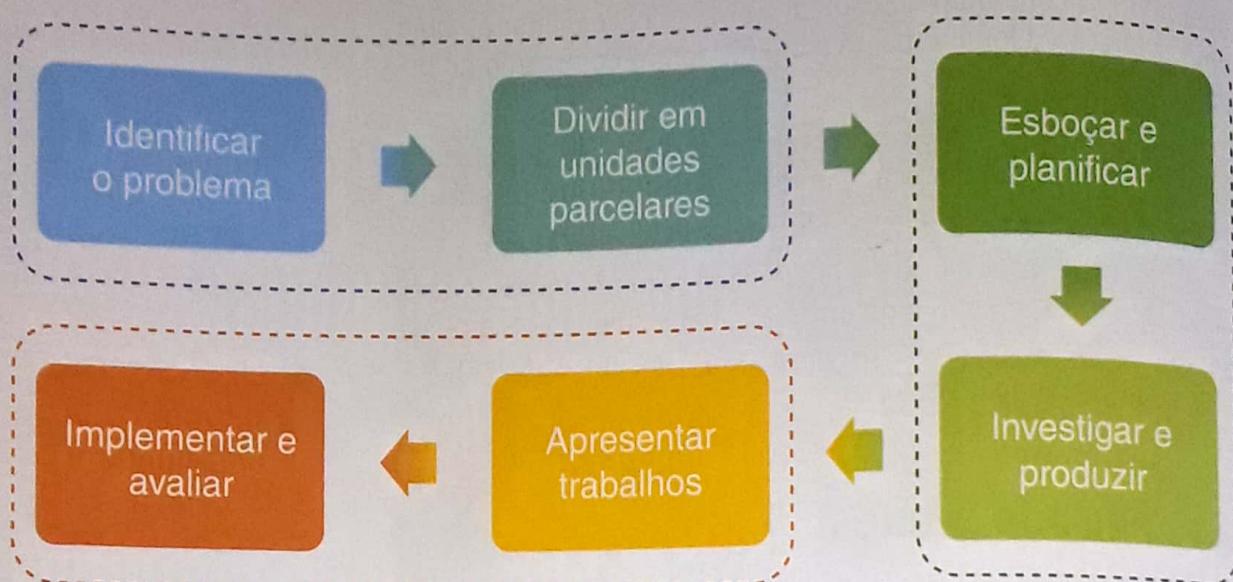


Fig. 2

Nota: As actividades e os documentos auxiliares que são facultados nesta unidade devem ser, respectivamente, efectuados e consultados na fase indicada, mas no momento temporal considerado como o mais adequado.

Os alunos podem considerar como documentos complementares os facultados no livro da 11.ª classe.

4.1. Identificar o problema

Esta fase corresponde à identificação e escolha da situação/problema que constituirá o tema do projecto a desenvolver pelo grupo. Esta escolha tem de ser fundamentada, isto é, deve ser explicada a razão ou razões da sua escolha.

A selecção do tema do trabalho de projecto é uma tarefa árdua que, por vezes, pode levar algum tempo até se encontrar um tema consensual e de interesse de todo o grupo de trabalho.

Os alunos deverão apresentar propostas de realização de projectos, em grupo ou individuais, devendo a escolha dos temas ser feita de acordo com os seus interesses, tendo em conta que poderão ser abordadas temáticas de outras disciplinas e outros assuntos relacionados com os grandes temas mundiais da actualidade.

Na ausência ou dificuldade na escolha de um tema por parte dos alunos, o professor deve procurar identificar temas, sugerindo-os. Pode, inclusive, optar por um tema único sobre o qual cada grupo de trabalho trabalhará uma determinada particularidade.

A definição de um tema de trabalho deve ser conseguida através da discussão entre todos os intervenientes, de forma a conseguir-se um consenso e a motivação de todos os alunos. Este envolvimento e motivação são fundamentais para o sucesso desta metodologia de aprendizagem.

Independentemente do tema escolhido, este deve:

- ser autêntico e real;
- ser relevante e significativo para todo o grupo de trabalho;
- estar enquadrado no meio social dos elementos do grupo tendo em conta os recursos a que têm acesso;
- ser exequível no espaço de tempo disponível para a sua execução;
- ser transversal com as várias áreas curriculares.

Realiza a Actividade n.º 1 da página 120:
Brainstorming

Consulta o Apoio 2 da página 131:
Como redigir um relatório

Realiza a Actividade n.º 2 da página 121:
Relatório de escolha do tema

2. Dividir em unidades parcelares

A identificação de problemas parcelares passa pela identificação de questões que levarão os alunos a desenvolver a sua capacidade de investigação, a identificar os recursos, métodos e técnicas a utilizar para encontrar as respostas que, posteriormente, serão tratadas e organizadas, levando os alunos à conclusão do trabalho.

Deve aos alunos escolher e dividir entre si as tarefas, bem como proceder à sua execução, estando subjacente a planificação das actividades. Quando se trata de um tema único para a turma, esta tarefa deve ser coordenada pelo professor e ser realizada antes da formação de grupos de trabalho, os quais devem ser formados, posteriormente, em função de cada tema ou problema a resolver.



Fig. 3

Realiza a Actividade n.º 7 da página 126:
Ajuda intergrupos

Realiza a Actividade n.º 8 da página 126:
Relatório intermédio

O trabalho deve ser flexível e reajustado permanentemente às necessidades verificadas pelos alunos, não podendo a relação planeamento/concretização do projecto ser encarada com rigidez.

Nesta fase sugerimos que sejam efectuados dois relatórios intermédios, em momentos que o professor considere importantes. Desta forma, quer os alunos quer o professor tomam consciência da forma como o projecto está a evoluir.

4.5. Apresentar trabalhos

Os produtos finais são apresentados à comunidade, podendo assumir formas diversas: relatórios, filmes, exposições, dramatizações, objectos, maquetas, etc.

Antes de elaborar o produto final, os alunos devem, em grupo, tratar a informação recolhida, procedendo à sua análise e discussão. Devem identificar se todas as tarefas foram correctamente realizadas de acordo com a planificação e justificar as discrepâncias existentes entre a planificação e a execução das tarefas. Devem, ainda, seleccionar a informação útil e necessária para justificar as conclusões a que chegaram e o trabalho que realizaram.

Após o tratamento dos dados, os alunos devem elaborar o relatório final seguindo as indicações do professor. Mais uma vez o aluno tem condições para aplicar os conhecimentos adquiridos quando estudou o módulo de processamento de texto, com a inclusão de objectos externos, tais como imagens e gráficos do Excel.

O relatório deve reflectir, de forma sucinta, clara e objectiva, o trabalho realizado.

Sugerimos que o relatório se divida em três partes essenciais:

- parte preliminar – capa, folha de rosto, agradecimentos (elemento facultativo), sumário, lista de abreviaturas;
- parte textual – introdução, corpo do relatório, conclusão;
- parte referencial – apêndices, bibliografia, anexos, etc.

Para tornares o teu trabalho mais apelativo, inclui imagens e gráficos relacionados com os conteúdos abordados no relatório mas em quantidade adequada.

Nesta fase, os grupos de trabalho terão a oportunidade de dar a conhecer o trabalho realizado através da sua apresentação à turma ou a toda a comunidade escolar.

Mais uma vez, as ferramentas informáticas têm um papel muito importante, não só no que respeita à apresentação do trabalho como também na sua divulgação.

Recorrendo a uma aplicação para a criação de apresentações, tal como o PowerPoint, os alunos têm os recursos necessários para realizarem a apresentação do trabalho de projecto através de diapositivos. Beneficiando das potencialidades destas aplicações, é possível criar apresentações atractivas e bem estruturadas, sendo uma ferramenta que permite auxiliar e guiar os alunos durante a apresentação do trabalho.

Para a divulgação do trabalho, os alunos podem ainda criar folhetins informativos, cartazes e brochuras. Sugerimos ainda a criação de um website para a divulgação do trabalho realizado.

Consulta o Apoio 7 da página 141:
Dez sugestões para elaborar um PowerPoint apelativo

5.6. Implementar e avaliar

No fim, todos os elementos envolvidos no trabalho devem elaborar um balanço, conduzindo este à avaliação do trabalho realizado pelos alunos.

A avaliação não deve incidir apenas sobre o trabalho final, mas deve ter um carácter contínuo, permitindo a avaliação de todo o trabalho realizado ao longo do projecto. Neste sentido, o professor, no decorrer do desenvolvimento do projecto, deve criar os próprios registos de avaliação, registando a evolução do trabalho, o envolvimento dos alunos, o cumprimento dos objectivos intermédios através da realização das tarefas planeadas e todas as restantes anotações que considere necessárias para uma avaliação justa e eficaz.

É importante que a avaliação dos trabalhos seja realizada em conjunto com todas as pessoas envolvidas no projecto, os alunos e os professores das áreas curriculares sobre as quais incidiu o trabalho de projecto.

A elaboração de relatórios é um bom método para, de um modo claro, objectivo e sucinto, poder descrever a situação à partida, o processo, os reajustamentos pontuais e a situação à chegada.

É importante realçar o papel que a auto-avaliação do projecto, feita pelos alunos, pode desempenhar e que os ajudará a reflectir sobre o seu próprio processo de aprendizagem e a constatar os progressos realizados.

Actividades

Actividade n.º 1 – Brainstorming

Sumário da actividade:

Brainstorming é uma maneira útil de começar a surgir novas ideias. Pode ser feito na turma, em pequenos grupos ou individualmente. Uma vez familiarizados com o processo, os alunos podem usar esta técnica sozinhos, quando ficam bloqueados, quando revêem o seu trabalho ou quando passam para uma nova fase.

Materiais:

- Um lugar para registar as ideias que todos possam ver (quadro, transparência, papel cenário ou cartões de colar)
- Cadernos de notas, cadernos de ideias

O que fazer?

- 1 O brainstorming começa com uma sessão de criação de ideias. Os alunos devem criar o máximo de ideias possível, sem avaliação ou censura. Nesta fase, a quantidade de ideias é mais importante do que a qualidade. Os alunos devem ser encorajados a incluir ideias estranhas, perspectivas novas e diversificadas, ideias que construíram em sugestões prévias e que funcionaram no passado. Pode ser designado alguém para registar as ideias à medida que surgem num lugar visível por todos. Esta sessão dura, geralmente, 5-10 minutos, mas pode ir até aos 20 minutos.
- 2 Nos mesmos grupos, ou como turma, os alunos devem discutir as ideias que formularam e trabalhar para as resumir numa lista ou num plano para as desenvolver. Podem recorrer a perguntas, tais como as seguintes, para guiar a sua discussão:
 - O que nos diz a nossa lista acerca do modo como vêm este assunto?
 - Quais as áreas ou interesses que estão ausentes desta lista?
 - Estas ideias enquadram-se nos nossos constrangimentos (recursos, tempo, acessibilidade)?
 - Podemos formular novas ideias analisando ou discutindo esta lista?
 - O que é que vamos escolher desta lista?
- 3 Os alunos podem querer organizar as ideias agrupando-as por categorias. Sugerir as seguintes três categorias: as imediatamente úteis, as de exploração futura e as novas abordagens ao tema.

Notas para os professores:

- O brainstorming pode ser usado ao longo do projecto, quando os alunos iniciam outra fase, estão bloqueados ou precisam de novas ideias. É uma boa ferramenta para os coordenadores do grupo que podem usá-la para focar o grupo ou criar novas ideias.
- Há muitas oportunidades de avaliação nas sessões de brainstorming, tanto para os alunos como para o professor. É uma boa oportunidade para o grupo e cada aluno reflectirem sobre o que fazer em seguida, para os alunos reflectirem nos relatórios e entradas do portefólio sobre o projecto e a actividade do grupo e, ainda, para colocarem questões ao professor.

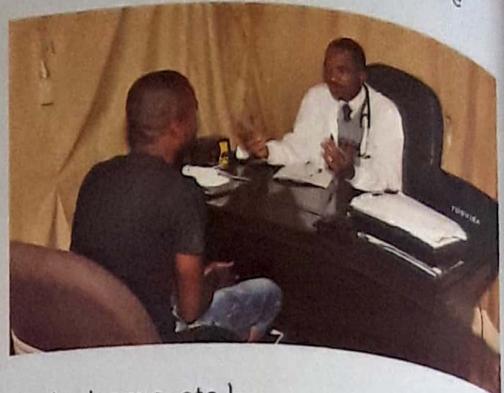
Projectos

Projecto n.º 1

Objectivo: "Criação de um gabinete de diálogo e aconselhamento"

Nas escolas existem alunos de vários níveis etários, com diversos tipos de problemas e com necessidade de esclarecimento em diferentes assuntos, tais como: vida sexual, relação com a família, prevenção e tratamento do uso de droga e/ou de alcoolismo, etc.

Assim, será muito importante existir nas escolas um gabinete com técnicos especializados que, em dias e horas diferentes, atendam os jovens para os esclarecerem sobre as suas dúvidas de modo a prevenir qualquer tipo de situação indesejada, aconselhar sobre as consequências nefastas do consumo de droga, promover a entreajuda dos alunos mais velhos aos mais novos, sensibilizando-os para os perigos dos comportamentos de risco, etc.



Para isto, cada grupo de trabalho deve:

- seleccionar um tema (por exemplo: vida sexual, combate ao uso de drogas, etc.);
- com a ajuda dos professores, contactar especialistas nestes temas que se disponibilizem a ir à escola, num horário a determinar, prestar esclarecimentos e outros serviços;
- usar uma sala para criar o gabinete, definir um horário e divulgar pela escola o novo tipo de serviço de que todos podem usufruir.

Desta forma espera-se que problemas associados a estes temas diminuam e que os jovens cresçam de forma mais saudável e esclarecida.

Projecto n.º 2

Objectivo: "Dar a conhecer o país em que vives"

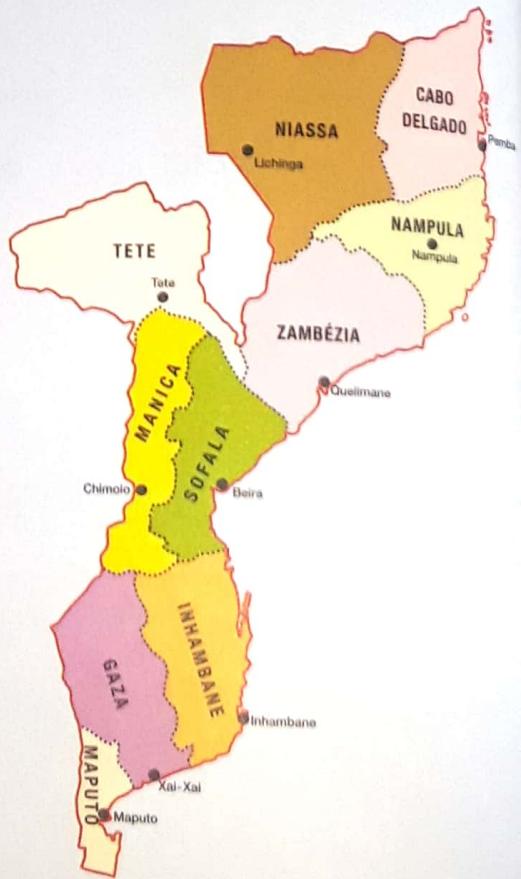
Este projecto tem como principal objectivo dar a conhecer aos alunos da escola e também aos professores, pais e outros interessados, a história, as características, a gastronomia, o clima, e outros aspectos típicos das várias cidades de Moçambique.

É natural que, vivendo numa determinada cidade do país, não conheças outras cidades e que sintas bastante curiosidade e interesse em conhecer.

Este projecto pressupõe a existência de um espaço, equipado com cadeiras, mesa e outro equipamento que aches necessário, como, por exemplo, um computador.

O grupo de trabalho deve efectuar pesquisas na Internet, realizar entrevistas, promovendo o uso do e-mail, entre outras estratégias, que possibilitem conhecer as principais características de diferentes cidades de Moçambique.

Com os dados recolhidos, os alunos poderão organizar apresentações (cartazes, panfletos, apresentações em PowerPoint, entre outros) que permitam descrever as cidades estudadas.



Actividade n.º 6 - Ficha de leitura

análise de documento escrito em jornal, revista, livro... (agrafar cópia do documento analisado).

Aluno: _____
Data: 20 - - -

N.º: _____ Ano/Turma: _____

Identificação do documento	
Autor/Entidade responsável	
Título onde foi obtido	
Data de publicação	
Outros dados bibliográficos	

Síntese do documento:

Tópicos/Assuntos do projecto com o(s) qual(is) é possível relacionar o documento:

- _____
- _____
- _____
- _____
- _____
- _____

Explicação da relação entre o documento e (pelo menos) um tópico do projecto:

Deverão promover duas sessões semanais e em cada uma delas apresentar uma cidade. Os intervenientes devem favorecer o diálogo e o debate, convidando pessoas que conheçam bem a cidade em análise ou respondendo a questões do público.

Os alunos deverão, junto do público, recolher propostas de cidades a estudar e anunciar as palestras de apresentação com panfletos elaborados pelo grupo de forma a despertar curiosidade nas pessoas e vontade de participar.

Desta forma promove-se um conhecimento mais amplo do país em que vives.

Projecto n.º 3

Objectivo: "Criar um produto original"

Um empreendedor é aquele que:

... sabe olhar, por exemplo, para um pneu, desenvolver a sua própria técnica e começar a fazer chinelos com esse material, tornando os seus produtos o sucesso do bairro."

Pois é, também tu podes ser um empreendedor!

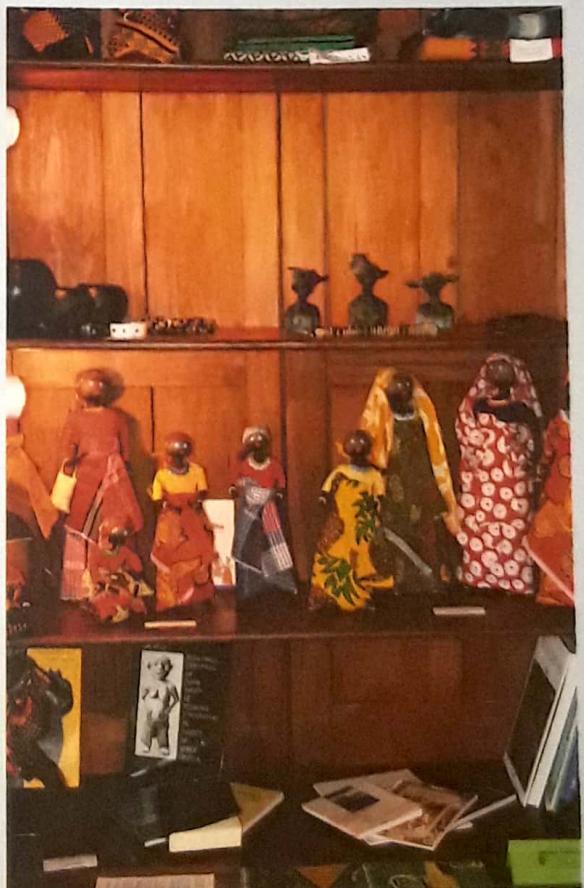
Já alguma vez olhas-te para objectos usados ou velhos e pensas-te que poderias torná-los úteis? Ou, então, a partir de uma qualquer matéria-prima, pensaste que podias produzir algo útil?

Por exemplo: com trapos podes fazer bonecas ou outro tipo de artesanato, em frascos podes plantar plantas, etc.

Este projeto deve ser desenvolvido em grupos em que cada grupo deve:

- começar por tentar identificar objectos – peças plásticas, cartões, tábuas, etc. – e, de forma criativa, pensar numa forma de transformá-los em objectos úteis;
- expor a ideia ao professor, mostrando um exemplar do material original e outro do objecto idealizado;
- depois da aprovação do professor, começar o fabrico dos objectos;
- um ou dois elementos do grupo devem elaborar panfletos publicitários promovendo o produto e informando o local e a data onde vai ser vendido (o dinheiro das vendas deve ser usado para adquirir material escolar – livros, cadernos, lápis e outros – para distribuir pelos alunos mais carenciados da escola e da comunidade);
- na data marcada todos os grupos devem promover uma "feira" na escola onde irão vender os seus produtos.

Claro que esta iniciativa deve ser mantida ao longo do tempo e poderão, inclusivamente, ir-se juntando outros "empreendedores", pois desta forma os desperdícios que iriam para o lixo são aproveitados e a comunidade consegue angariar dinheiro para ajudar os mais necessitados.



Apoio 1 – Regras de trabalho em grupo

Para que o trabalho em grupo tenha sucesso, seja produtivo e exequível, é importante a observação de algumas regras fundamentais, a saber:

- conhecer os diferentes papéis necessários ao funcionamento do trabalho em equipa;
- definir as tarefas/competências inerentes a cada função;
- atribuir os papéis de acordo com as características de cada elemento. É, contudo, importante que todos os elementos do grupo desenvolvam experiência nos diversos papéis, pelo que se recomenda a sua alternância.

Papel	Tarefas	Competências
Moderador	<ul style="list-style-type: none">- Organização da participação individual- Gestão do tempo	<ul style="list-style-type: none">- Disciplinar as intervenções- Intervir no momento oportuno- Ser firme na tomada de decisão- Cronometrar o tempo
Secretário	<ul style="list-style-type: none">- Registo global provisório da colaboração individual dos membros	<ul style="list-style-type: none">- Elevada concentração- Rapidez de escrita- Capacidade de síntese- Seleccionar/Sistematizar a informação útil
Secretário TIC	<ul style="list-style-type: none">- Processamento de texto (Word) em suporte papel/informático	<ul style="list-style-type: none">- Correcção ortográfica- Saber compilar e estruturar a informação por itens/campos- Domínio das ferramentas informáticas

No entanto, compete a todos os elementos colaborar/contribuir activamente para a consecução dos objectivos pretendidos. O porta-voz do grupo deve ser rotativo, de modo a garantir que todos os membros estão integrados das fases do projecto.

Regras elementares

- Todos os elementos devem estar receptivos à opinião do outro e saber fazer cedências.
- Os membros do grupo devem procurar tomar decisões por unanimidade.
- Encarar a crítica fundamentada como condição necessária à qualidade do trabalho.
- Perspectivar o conflito entre membros de forma positiva:
 - como expressão da criatividade/diversidade pessoais;
 - como oportunidade para desenvolver competências de relacionamento interpessoal.
- É imprescindível a colaboração de todos os membros de modo a garantir resultados optimizados.
- Cada membro faz uma avaliação:
 - do produto final (resultados obtidos);
 - da sua prestação individual (auto-avaliação);
 - do desempenho de cada elemento da equipa (heteroavaliação).

Apoio 2 – Como redigir um relatório

1. Considerações gerais

1.1. Orientações de formatação

• Os relatórios intermédios (relatórios do processo) quer o relatório final devem seguir as seguintes

formas formais:

• Papel branco, formato A4 (210 x 297 mm)

• Orientação – vertical

• Margens – Superior – 2,5 cm

• Inferior – 2,5 cm

• Esquerda – 3 cm

• Direita – 3 cm

• Tipo de letra – Times New Roman ou Arial

• Tamanho – 12 para a letra do texto e 8 para as notas de rodapé

• Espaçamento entre linhas – 1,5

• Parágrafo – recuo de 12,5 mm na primeira linha dos parágrafos

• Paginação – a folha de rosto não costuma ser numerada mas conta para numeração (página um);

• As restantes páginas são numeradas sucessivamente com numeração árabe.

1.2. Redacção

• Utilização da primeira pessoa do plural (ex.: nós fizemos...) ao longo de todo o relatório, com exceção

obvia de quando se descreve a participação individual de cada elemento do grupo.

• As frases e os parágrafos não devem ser demasiado extensos.

• No caso de haver necessidade de incluir citações e notas de rodapé, estas devem ser colocadas na parte

inferior da página a que se referem.

• As ideias devem ser apresentadas com clareza, rigor científico e bem estruturadas.

1.3. Aspectos a ter em conta na elaboração dos relatórios

• Correcção e clareza da expressão escrita

• Rigor científico

• Correcção e clareza dos raciocínios

• Criatividade

2. Formalmente um relatório deve ser constituído por:

2.1. Parte preliminar (capa, folha de rosto, agradecimentos (elemento facultativo), sumário, lista de abreviaturas)

2.2. Parte textual (introdução, corpo do relatório, conclusão)

2.3. Parte referencial (apêndices, bibliografia, anexos...)

2.1. Parte preliminar

• Capa do relatório/folha de rosto¹ deve conter:

• Nome da escola

• Título do trabalho

• Identificação dos autores

• Identificação do professor

¹ Num relatório de pequena dimensão não é necessária capa, constando da folha de rosto os seguintes dados: nome dos autores, título e data.

Apoio 6 – Como realizar uma entrevista

Entrevista por entrevista

A entrevista é um método de recolha de informações que consiste em conversas orais, individuais ou de grupo, com várias pessoas seleccionadas cuidadosamente, cujo grau de pertinência, validade e fiabilidade é analisado na perspectiva dos objectivos da recolha de informações" (Ketele, 1999: 18).

através de um questionamento oral ou de uma conversa, um indivíduo ou um informante-chave pode ser interrogado sobre os seus actos, as suas ideias ou os seus projectos. Previamente, a entrevista carece de um propósito (tema, objectivos e dimensões) bem definido e é essencial ter uma imagem do entrevistado, procurando caracterizar sucintamente a sua pessoa. De seguida, selecciona-se a amostra dos indivíduos a entrevistar segundo um método representativo da população ou de oportunidade.

O guião de entrevista é um instrumento para a recolha de informações na forma de texto que serve de base à realização de uma entrevista propriamente dita. O guião é constituído por um conjunto ordenado ou não de questões que podem ser de três tipos distintos:

- questões abertas (resposta livre);
- semiabertas (parte da resposta fixa e outra livre);
- fechadas (resposta fixa).

Deve incluir uma indicação da entidade e/ou pessoa, data, local e título. Um texto inicial apresenta a entrevista e os seus objectivos, devendo ser lido ao entrevistado. O guião ainda pode conter notações que auxiliam a condução da entrevista (tempo previsível de resposta, palavras-chave de resposta, questões para aprofundamento do tipo "pode dizer-me mais acerca deste assunto?", etc.).

Considerando a entrevista como um acontecimento social, não se devem desprezar as características onde decorre e a influência no entrevistado. Tem-se em consideração os diversos factores que podem interferir com as respostas:

- situação (recompensa, lugar, tempo...);
- características do entrevistado (memória, motivação...);
- características do entrevistado (grupo étnico, aspecto...);
- papel do entrevistador (vocabulário, papel, mensagens corporais...);
- características da mensagem (se é compreendida...).

Antes da realização da entrevista, deve-se efectuar um pré-teste, podendo, eventualmente, ajustar o guião da mesma. Depois de registada em papel ou em suporte magnético ou digital, a informação recolhida pela entrevista é tratada com vista à sua análise e à redacção das conclusões.

Etapas na elaboração do guião de entrevista

- 1 Descrição do perfil do entrevistado (nível etário, escolaridade, nível sociocultural, personalidade...).
- 2 Seleção da população e da amostra de indivíduos a entrevistar.
- 3 Definição do propósito da entrevista (tema, objectivos e dimensões).
- 4 Estabelecimento do meio de comunicação (oral, escrito, telefone, e-mail...), do espaço (sala, jardim...) e do momento (manhã, duração...).

5

Discriminação dos itens ou características para o guião:

- 5.1. Elaborar perguntas dos itens, de acordo com o definido nos pontos anteriores.
- 5.2. Considerar as expectativas do entrevistador.
- 5.3. Considerar as possíveis expectativas dos leitores/ouvintes.
- 5.4. Formular perguntas abertas (O que pensa de...?) e fechadas (Gosta de...?).
- 5.5. Evitar influenciar as respostas.
- 5.6. Apontar alternativas para eventuais fugas à pergunta.
- 5.7. Estabelecer o número de perguntas e proceder à sua ordenação dentro de cada dimensão.
- 5.8. Adequar as perguntas ao entrevistado, seleccionando um vocabulário claro, acessível e rigoroso (sintaxe e semântica).

6

Produção do guião com boa apresentação gráfica:

- 6.1. Redigir o cabeçalho com identificação (instituição, proponentes, título, data).
- 6.2. Incluir uma apresentação sucinta da entrevista, incluindo os objectivos.
- 6.3. Alinhar as perguntas na vertical e com espaçamento ajustado.
- 6.4. Utilizar tipo de letra legível, parágrafo justificado, margens da página com 2 cm e, eventualmente, imagens à direita do texto.

7

Validação da entrevista pela análise e crítica de personalidades relevantes.

Procedimentos durante a entrevista

1

Início com explicação da entrevista:

- 1.1. Esclarecimento do que pretende o entrevistador e do objectivo da entrevista.
- 1.2. Assegurar a confidencialidade do entrevistado e das suas respostas.
- 1.3. Ressaltar a necessidade da colaboração do entrevistado, sem tolhimento de qualquer ordem.

2

Criação de um ambiente agradável para a realização da entrevista:

- 2.1. Verificar que o espaço/local da entrevista favorece a descontração do entrevistado (temperatura, luz, móveis...).
- 2.2. Manter uma distância audível entre o entrevistado e o entrevistador (1 a 2 metros).
- 2.3. Em casos de entrevista a um grupo, pode ser vantajoso subdividi-lo em pequenos grupos.
- 2.4. Verificar se existem condições de privacidade do entrevistado.
- 2.5. Permitir que o entrevistado mantenha o controlo da entrevista.

2 Análise das respostas às questões fechadas através de medidas estatísticas:

- 2.1. Determinar as percentagens das opções de resposta em cada ítem.
- 2.2. Calcular a moda das variáveis qualitativas.
- 2.3. Calcular as medidas estatísticas (média, mediana, desvio-padrão, desvios...) das variáveis quantitativas.

3 Determinação de um conjunto de categorias (temas, tópicos...) das respostas às questões abertas:

- 3.1. Elaborar uma grelha de registo das categorias apontadas em cada entrevistado.

Entrevistados	Dimensões de análise	Tema 1	Tema 2	Tema 3	Tema 4
Entrevistado 1					
Entrevistado 2					

- 3.2. Descobrir padrões pela leitura dos dados da entrevista (comparação, ordenação, relação com características individuais – idade, sexo...).

4 Elaboração do relatório:

- 4.1. Explicitar a metodologia do inquérito, incluindo a selecção da população e da amostra e a justificação, a elaboração e a validação do instrumento da recolha de dados.
- 4.2. Descrever a recolha e o tratamento dos dados.
- 4.3. Apresentar a análise dos dados (tabelas, gráficos, resultados estatísticos, semelhanças e diferenças nas respostas dos entrevistados, padrões de declarações e correspondência com características individuais), acrescentando as respostas às questões porque acontece, quando acontece, quando não acontece...).
- 4.4. Explicitação das conclusões da entrevista (síntese, resultados, reflexões, implicações, sugestões...).

5 Disponibilização dos materiais utilizados (anexos, bibliografia, dados (?)...).

Apoio 9 – O cronograma

O cronograma é a previsão de tempo que será gasto na realização do trabalho de acordo com as actividades a serem cumpridas. As actividades e os períodos serão definidos a partir das características de cada pesquisa e dos critérios determinados pelo autor do trabalho.

Os períodos podem estar divididos em dias, semanas, quinzenas, meses, bimestres, trimestres, etc. Estes serão determinados a partir dos critérios de tempo adoptados por cada pesquisador.

O cronograma do projecto separa as etapas do mesmo de acordo com o tempo integral de concretização, isto para garantir que o projecto é concluído a tempo.

Um cronograma de projecto é criado para:

- coordenar tarefas com outras actividades organizacionais que estejam a ocorrer fora do projecto;
- coordenar tarefas e dependências dentro do próprio projecto;
- determinar a ordem das actividades, garantindo que as primeiras coisas são feitas primeiro;
- definir o que é feito e quando, mas, para determinar se o cronograma é viável, é necessário definir quem fará o quê e quando.

Em projectos complexos, o cronograma não deve ser encarado como um caminho fixo, mas, na verdade, como uma forma de organizar, comunicar e comparar expectativas com a realidade.

Componentes do cronograma

Um cronograma deve incluir:

- datas planeadas para o início e esperadas para o fim de todas as actividades do projecto;
- principais marcos e/ou eventos-chave para implementar o projecto;
- dependências e sequência das actividades.

Exemplo:

	ACTIVIDADES / PERÍODOS	1	2	3	4	5	6	7	8	9	10
1	Levantamento de literatura	X									
2	Montagem do projecto		X								
3	Colecta de dados			X	X	X					
4	Tratamento dos dados				X	X	X	X			
5	Elaboração do relatório final						X	X	X		
6	Revisão do texto									X	
7	Entrega do trabalho										X

8 Importa outras imagens e gráficos

Não limite a tua apresentação ao que é oferecido pelo PowerPoint. Utiliza imagens e gráficos externos para maior diversidade e um aspecto mais agradável, incluindo vídeo. "Estes ajudam a divertir, transmitem uma mensagem e descontraem a audiência."

9 Distribui folhetos no final e nunca durante a apresentação

Penso que nenhum orador está interessado em falar para uma audiência que está ocupada a ler um resumo dos seus comentários. Se não for imperativo que as pessoas sigam os folhetos durante a apresentação, espera até terminar para os distribuir, caso necessites.

10 Verifica cuidadosamente a apresentação antes de começar

Nunca percas a perspectiva da audiência. Quando terminares de delineares os teus diapositivos de PowerPoint, assume que és apenas uma das pessoas que vai ouvir os teus comentários e faz uma revisão. Se algo for pouco apelativo e possa servir para distrair ou confundir a audiência, corrige cuidadosamente. É bastante provável que a tua apresentação fique melhor depois de revista.

Apoio 8 – O que é um portefólio?

O portefólio de um aluno é, no fundo, uma pasta ou um *dossier* contendo elementos importantes do trabalho que ele realizou na disciplina ao longo de um ano lectivo ou num determinado período do ano.

O portefólio deve conter:

- os principais trabalhos do aluno;
- relatórios que elaborou;
- descrição de problemas que resolveu;
- explorações e investigações em que esteve envolvido;
- testes realizados;
- etc.

Esses elementos devem estar acompanhados dos comentários que o professor e o próprio aluno foram fazendo a propósito das diversas actividades realizadas. O portefólio deve reflectir globalmente o percurso do aluno e pode constituir um testemunho valioso o facto de se guardar, por exemplo, o trabalho preferido, os comentários feitos numa reflexão pessoal sobre uma actividade ou as respostas dadas a um questionário sobre as aulas.

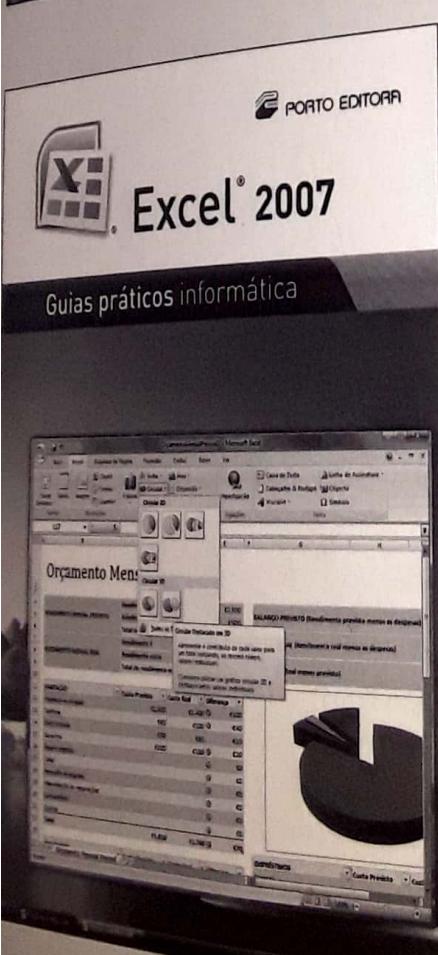
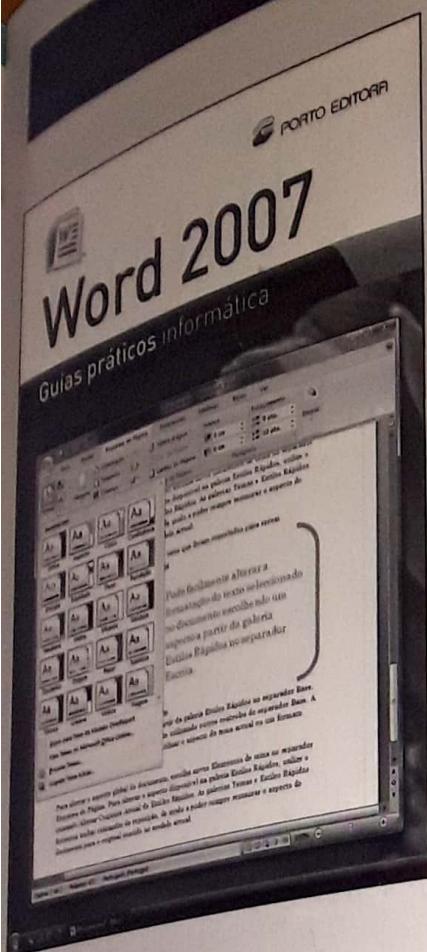
A selecção do material a incluir no portefólio deve ser da responsabilidade conjunta do aluno e do professor. (...) Para o aluno, pode contribuir para desenvolver o sentido de responsabilidade, de organização e os hábitos de reflexão. Do ponto de vista do professor, ajudá-lo-á a ter uma visão global do trabalho do aluno e a focar sobretudo na sua evolução mais do que em aspectos isolados ou pontuais daquilo que ele fez.

O seu valor, nomeadamente do ponto de vista da auto-avaliação, pode estar na selecção e organização do material que é incluído e na justificação que o aluno apresenta para a escolha desse material. Por isso mesmo, será útil destinar periodicamente algum tempo e atenção à tarefa específica de organizar o portefólio, uma tarefa que requer, ela própria, orientação da parte do professor.

Propõe-se que o portefólio seja estruturado da seguinte forma:

- índice;
- introdução, descrevendo e justificando o seu conteúdo;
- documentos criados.

Outros títulos de apoio



Guias Práticos: Word

O Microsoft® Office System 2007 inclui alterações fundamentais que se traduzem numa nova abordagem de comunicação com o utilizador. Excelente para quem toma agora contacto com os módulos do Office, estas alterações podem requerer alguma reaprendizagem para quem já estava familiarizado com as aplicações.

Facilitar a migração dos utilizadores de versões anteriores do Office – e, neste caso concreto, do Word – para a versão 2007, é um dos principais objectivos deste **Guia prático**.

Guias Práticos: Excel

O Excel® 2007 foi redesenhado para poder localizar e utilizar as capacidades do programa mais facilmente. O aspecto e funcionamento globais foram aperfeiçoados. Para além disso, tecnologias novas permitem-lhe “procurar, seleccionar e clicar” em vez de ter de seleccionar opções em caixas de diálogo complicadas. Também poderá obter melhores resultados e mais rapidamente tirando partido dos abrangentes conjuntos de funcionalidades apresentados na nova interface do utilizador da aplicação.

Glossário

1GL – (Linguagem de primeira geração) Linguagem de programação em que as instruções são expressas em código máquina (código binário composto apenas pelos dígitos 0 e 1).

2GL – (Linguagem de segunda geração) Linguagem de programação que, apesar da evolução relativamente à primeira geração, ainda exige que o programador conheça as características da arquitectura do microprocessador. Ex.: Assembly.

3GL – (Linguagem de terceira geração) Linguagem de programação desenvolvida para ser facilmente entendida pelo ser humano e, portanto, muito mais simples de utilizar por qualquer programador. Ex.: Pascal, Basic, C, C++.

4GL – (Linguagem de quarta geração) Linguagem de programação de alto nível adequada ao desenvolvimento de programas complexos. Ex.: SQL, Progress 4GL.

Administrador de sistema – Responsável pelo correcto funcionamento de um sistema informático (base de dados, rede de computadores, etc.).

Algoritmo – Sequência finita de instruções bem definidas e não ambíguas. Um algoritmo não representa, necessariamente, um programa de computador, podendo descrever os passos necessários para realizar uma tarefa.

Antivírus – Programa cuja função é detectar e neutralizar o efeito nocivo de um vírus.

Backup – Cópia de segurança dos ficheiros existentes num dispositivo de armazenamento para outro, possibilitando a sua reposição em caso da perda ou necessidade de correcção.

Base de dados – Conjunto de informação relacionada.

Bug – Erro detectado no funcionamento de um programa (software) ou de um componente informático (hardware).

Correio electrónico – Forma de comunicação baseada no envio e recepção de mensagens, normalmente através de uma rede de computadores.

Cracker – Pessoa que ultrapassa barreiras de segurança, em sistemas informáticos, de forma ilegal ou sem ética.

Data Warehouse (DW) – Sistema de gestão de bases de dados com funcionalidades que permitem o processamento de grandes quantidades de dados.

E-mail – (Electronic mail) O mesmo que correio electrónico.

FTP – (File Transfer Protocol) Método de transferência de dados entre computadores.

Hacker – Pessoa que tem como desafio ultrapassar barreiras informáticas, acedendo a sistemas informáticos, sites ou páginas de acesso restrito, sem autorização.

ISP – (Internet Service Provider) Empresa que permite o acesso de um computador à Internet.

Linguagem de programação – Conjunto de regras utilizadas na criação de um programa de computador.

Password – Palavra de acesso. Código usado para aceder a um sistema.

Programa informático – Conjunto de instruções que descrevem tarefas a serem realizadas pelo computador.

Programação – Processo de escrita, teste e manutenção de um programa de computador.

Rede de computadores – Conjunto de computadores (ou apenas processadores) capazes de trocar/partilhar informações e recursos, interligados por um subsistema de comunicação.

Servidor – Computador central que gera a informação a que outros computadores, que a ele estão ligados em rede, podem aceder.

SGBD – (Sistema de Gestão de Bases de Dados) Programas responsáveis pela gestão de bases de dados. Ex.: Microsoft Access, OpenOffice Base, Oracle, MySQL.

Spyware – É uma forma de vírus informático que normalmente acede ou regista informação do utilizador, sem autorização.

Vírus informático – Programa malicioso, que danifica ou altera o normal funcionamento de um sistema informático. Pode ser transmitido de máquina para máquina sem que o utilizador se aperceba.

07 - Dez sugestões para elaborar um PowerPoint apelativo

1 Defende a tua imagem com material apelativo

De certa forma, a facilidade de utilização do PowerPoint poderá ser a tua maior desvantagem. No entanto, apesar da facilidade de criação de diapositivos e gráficos apelativos, não te esqueças que o PowerPoint não é autónomo. O público veio para te ouvir e não para ficar apenas a olhar para imagens projectadas num ecrã. Cria um programa forte de PowerPoint, mas garante que os teus comentários verbais são igualmente convincentes. "O PowerPoint não faz apresentações — o PowerPoint faz diapositivos"; lembra-te que estás a criar diapositivos de suporte a uma apresentação oral.

2 Não compliques

Provavelmente, já todos assistimos a apresentações de PowerPoint e outras onde o orador parecia querer "casar-se com o programa". Afinal, era claro que ele se tinha apaixonado por todas os botões, efeitos especiais e técnicas disponíveis. Mas as apresentações em PowerPoint mais eficazes são simples — gráficos de fácil compreensão e imagens que reflectem o discurso do orador. Alguns especialistas sugerem o máximo de cinco palavras por linha e um máximo de cinco linhas por cada um dos diapositivos. "Não prejudiques o trabalho abusando das palavras e gráficos.", "Será mesmo necessário ter tanta informação no ecrã?"

3 Reduz ao mínimo os números nos diapositivos

O fascínio do PowerPoint é a capacidade de transmissão de ideias e o apoio conciso aos comentários do orador. Isto é difícil de conseguir através de um emaranhado de números e estatísticas. Quase todas as apresentações PowerPoint mais eficazes não sobrecarregam a audiência com demasiados números. Deixa esses números para mais tarde, para serem digeridos nos folhetos distribuídos no final da apresentação. Se pretendes dar ênfase às estatísticas no PowerPoint, considera a utilização de um gráfico ou de uma imagem para transmitir a ideia.

4 Não te limites a utilizar o PowerPoint para discursar como um papagaio

Um dos hábitos mais predominantes e prejudiciais dos utilizadores do PowerPoint é a leitura simples para o público da apresentação visual. Para além de se tornar redundante — para além de mudar os diapositivos, o que é que estás lá a fazer? —, consegues tornar tediosas as apresentações mais apelativas. O PowerPoint funciona melhor se utilizares comentários verbais explicativos e que encorajem o debate, em vez de te limitares a repetir o que é apresentado no ecrã. "Mesmo utilizando o PowerPoint, tens de estabelecer contacto visual com a audiência.", "As pessoas não estão lá para te ver de costas."

5 Escolhe o momento adequado para falar

Outras armadilhas potenciais são os comentários verbais dos oradores coincidirem precisamente com a apresentação de um novo diapositivo do PowerPoint. Isto apenas faz com que a audiência divida as atenções. Um programa de PowerPoint bem organizado mostra um novo diapositivo, dá à audiência a hipótese de o ler e digerir e, em seguida, é acompanhado pelos comentários verbais que alargam e explicam mais detalhadamente o que está no ecrã. "É uma questão de *timing*.", "Nunca fales por cima dos teus diapositivos."

6 Dá-lhes descanso

Mais uma vez, o PowerPoint é mais eficaz como acompanhamento visual da fala. Os utilizadores experientes do PowerPoint não se importam de deixar o ecrã em branco ocasionalmente. Isto serve não apenas para dar uma pausa visual à audiência como também para chamar a atenção para a comunicação verbal, como acontece num debate em grupo ou numa sessão de perguntas e respostas.

7 Utiliza cores vibrantes

Um contraste apelativo entre palavras, gráficos e o fundo pode ser muito eficaz na transmissão da mensagem e de emoções.

- Favorecimento das respostas pertencentes ao entrevistado:
- 3.1. Mostrar compreensão e simpatia pelo entrevistado.
 - 3.2. Usar um tom informal, de conversa, mais do que de entrevista formal.
 - 3.3. Apresentar a questão oralmente e por escrito (combinar as duas linguagens).
 - 3.4. Começar com questões fáceis de responder (para pôr o entrevistado à vontade).
 - 3.5. Pedir ao entrevistado para dizer em voz alta o que está a pensar, o que pensou em fazer, se está com alguma dificuldade na resposta...
 - 3.6. Evitar influenciar as respostas pela entoação ou destaque oral de palavras.
 - 3.7. Pedir exemplos de situações, de pessoas ou de objectos que o auxiliem a exprimir-se.
 - 3.8. Apresentar uma questão de cada vez.
 - 3.9. O entrevistador explicita aceitação pelas opiniões do entrevistado (entrevista diferente de exame).

4. Registo de tudo o que o entrevistado diz:

- 4.1. Previamente, verificar suportes de registos (papel, fita, pilhas, captação do som...).
- 4.2. Antes de iniciar a entrevista, pedir autorização ao entrevistado para fazer a gravação.
- 4.3. Registar com as mesmas palavras do entrevistado, evitando resumi-las.
- 4.4. Anotar, se possível, gestos e expressões do entrevistado.

5. Gestão do tempo de conversação:

- 5.1. Duração de cerca de 25 minutos (?).
- 5.2. Terminar antes do tempo previsto se o ambiente se tornar demasiado constrangedor.

6. Finalização da entrevista como começou: num ambiente de cordialidade, para que o entrevistador possa voltar (se necessário) para obter novos dados.

Análise dos dados recolhidos pela entrevista

1. Verificação dos requisitos dos dados fornecidos pelo entrevistado:

- 1.1. *Validade* – comparar os dados com uma fonte externa.
- 1.2. *Relevância* – importância em relação aos objectivos.
- 1.3. *Especificidade e clareza* – referência com objectividade a dados, datas, nomes, locais, percentagens, prazos, etc.
- 1.4. *Profundidade* – relacionado com sentimentos e lembranças do entrevistado, sua intensidade e intimidade.
- 1.5. *Extensão* – amplitude da resposta.

Apoio 5 – Como fazer um inquérito

Para que um inquérito esteja bem realizado deve ser efectuado um trabalho de preparação que se traduz na construção da sua ficha técnica.

A ficha técnica do inquérito deve incluir os seguintes itens:

- **Objectivos:** os objectivos direcionam a procura de informação e ajudam à escolha das questões e à definição dos inquiridos. Devemos perguntar: «Vamos investigar o quê?». Deve ser previamente definido para que queremos aquela informação. Deve ser escolhido o tema do inquérito e até formulada a hipótese de partida (Será que as pessoas pensam...? O que é que...?). Não interessa recolher dados que não estejam relacionados com o projecto.
- **Amostra:** como é constituída pelas pessoas junto das quais vamos buscar a informação, deve ser caracterizada em termos quantitativos (número de inquiridos) e qualitativos (idade, profissão, sexo, etc.). Deve ser representativa, ilustrando a população-alvo e ter uma dimensão suficiente para permitir generalizações.
- **Conteúdos:** devem ser *listados* os assuntos acerca dos quais queremos obter informação (opiniões, dados, etc.), através da elaboração da lista de informações que necessitamos ter para fundamentar e/ou investigar um aspecto do projecto.
- **Objectivos das questões:** devem ser *explicados* e justificados através da tua relação com o desenho/visão do projecto. Os objectivos também definem o tipo de questão a formular: se é aberta ou fechada, se é qualitativa (muito, pouco), de frequência (às vezes, nunca...), etc.
- **Tratamento de dados:** deve ser previamente definido se além do tratamento quantitativo há lugar a análises de conteúdo/significação, etc. Este aspecto também auxilia a definição do tipo de questões a formular.

As questões do inquérito devem obedecer a algumas regras:

- as perguntas devem ser gerais;
- as perguntas devem ser claras e breves;
- o mesmo problema deve ser investigado através de perguntas diferentes (para se cruzar a informação, confirmando-a ou não);
- as questões devem partir das mais gerais para as mais restritas (ex.: Gosta de...? De que tipo?);
- o questionário deve motivar os inquiridos a preenchê-lo (não pode demorar muito tempo: daí as respostas mais previsíveis serem apresentadas de modo que só se solicite uma cruz; deve deixar-se em aberto outras respostas, incluindo a opção "outro": _____).

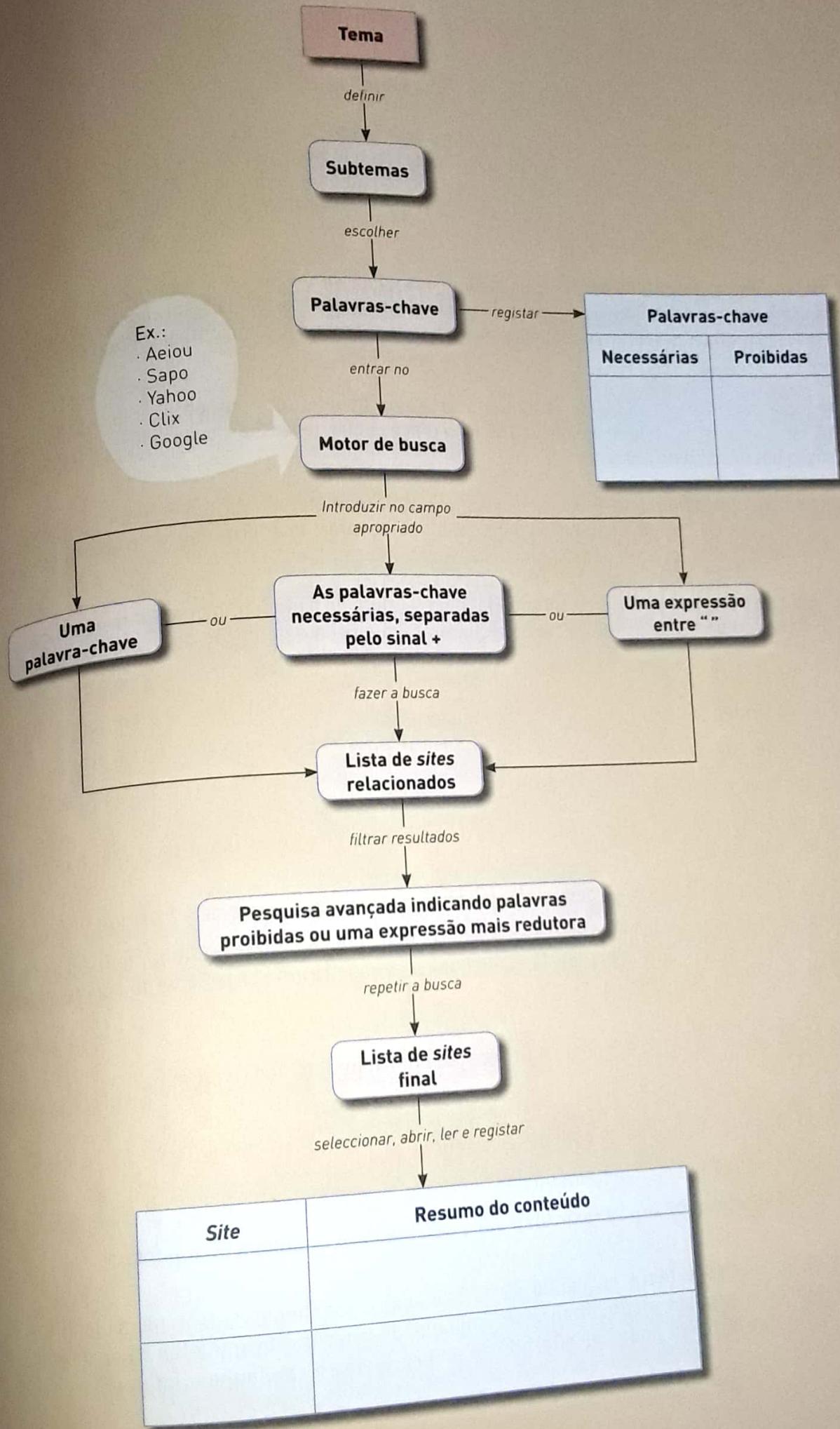
Definir o modo de inquérito tendo em conta a verificação ou tratamento dos resultados.

Ex.:

- Entrevista gravada: as respostas podem ser extensas.
- Questionário fechado: responder sim ou não, ou então às vezes, nunca, etc.
- Questionário aberto: aceitam-se respostas mais detalhadas e redigidas livremente.

É aconselhável fazeres um **pré-teste**, aplicando a 2 ou 3 colegas o questionário, de modo a identificar perguntas problemáticas (ambíguas, incompreensíveis, etc.) que devem ser reformuladas antes da aplicação. A ordem das perguntas também pode ser alterada se no pré-teste se revelou que influenciava o tipo de respostas.

Apoio 4 – Pesquisar na Web



3.ª Aplicação do inquérito

Esta fase corresponde ao trabalho no terreno. No questionário auto-administrado ou de administração directa, o inquirido regista as suas respostas. No questionário de administração indirecta, é o inquiridor que formula as perguntas e regista as respostas do inquirido.

4.ª Análise dos resultados

Nesta fase procede-se à codificação das respostas, ao tratamento dos dados e à elaboração das conclusões.

5.ª Apresentação dos resultados

Os resultados podem ser apresentados sob a forma de um relatório. A apresentação dos resultados do inquérito à comunidade pode assumir diferentes formas.

2 Entrevistas

A entrevista é uma técnica de investigação que permite recolher dados utilizando a comunicação verbal. Distinguem-se, geralmente, três tipos de entrevista:

1.º Entrevista não directiva ou livre

O entrevistador apresenta o tema permitindo ao entrevistado desenvolver livremente o seu pensamento.

2.º Entrevista directiva ou estruturada

Este tipo de entrevista permite recolher informação de uma forma estandardizada, dado que as pessoas respondem a um conjunto de perguntas idênticas.

3.º Entrevista semidirectiva ou semiestruturada

Os entrevistados orientam-se por um guião de temas que serão abordados livremente sem obedecer a uma ordem determinada.

A entrevista poderá ser registada através de meios audiovisuais (gravador ou vídeo). A sua apresentação pode ser feita por escrito ou através da reprodução do registo audiovisual.

3 Observação

Nas diferentes fases de pesquisa, o recurso à observação assume diferentes formas, de acordo com:

1.º Grau de inserção do observador no grupo ou na situação a observar

2.º A intencionalidade e organização da observação

Podemos distinguir a observação ocasional (que, não obedecendo a nenhuma regra, pode ocorrer em qualquer circunstância da vida quotidiana do observador) da observação sistemática (que é intencional e delimita o campo de observação). Uma observação mais aprofundada implicará a inserção do observado no grupo a observar – observação participante.

Apoio 3 - Técnicas de investigação

No desenvolvimento de um projecto, temos de recorrer a diferentes técnicas de investigação. Sinteticamente, apresentamos as técnicas a que recorreremos com maior frequência.

I - TÉCNICAS DOCUMENTAIS

A análise de textos, fornecidos pelos professores ou recolhidos pelos alunos em pesquisa bibliográfica, constitui uma importante fonte de informação nas diferentes fases de desenvolvimento do projecto.

Informações várias podem ser recolhidas em instituições públicas e privadas – arquivos, autarquias, associações profissionais, culturais, Instituto Nacional de Estatística, jornais, Internet, etc.

Para além de documentos escritos, há uma grande diversidade de outros materiais de informação: filmes, fotografias, diapositivos, DVD, gravações áudio, desenho, pintura...

II - TÉCNICAS NÃO DOCUMENTAIS

I - Inquérito por questionário

O inquérito por questionário é uma das técnicas de recolha de dados mais acessível. É constituído por uma série de perguntas organizadas segundo uma determinada ordem, dirigidas a um conjunto de indivíduos.

Um inquérito por questionário desenvolve-se em várias fases:

I.º Planeamento do inquérito

Em primeiro lugar será necessário definir o objecto de estudo, o tipo de informação que se pretende obter. Em função dos objectivos, deverão ser formuladas as hipóteses explicativas que irão orientar a construção do inquérito.

Ainda nesta fase proceder-se-á à delimitação da população ou universo do inquérito e ao cálculo de uma amostra representativa dessa população.

2.º Preparação do inquérito

Nesta fase proceder-se-á à redacção do questionário:

- Deve iniciar-se com uma apresentação onde sejam explicitados os objectivos a que se destina.
- Deve ser breve, apresentar um aspecto gráfico cuidado e utilizar uma linguagem clara e acessível aos inquiridos.

Na elaboração do questionário pode recorrer-se a dois tipos de perguntas:

• **Questões abertas** – os inquiridos respondem livremente à pergunta. Neste caso, o tratamento da informação é mais difícil, mas os dados obtidos são mais ricos.

• **Questões fechadas** – os inquiridos fazem uma escolha entre várias respostas que lhes são apresentadas. A principal vantagem reside no facto de o tratamento dos resultados ser facilitado pela codificação e normalização da informação.

A principal limitação das questões fechadas prende-se com a menor profundidade de informação. Para evitar o inconveniente de uma e outra opção, poder-se-á construir um inquérito com questões abertas e fechadas.

- Data
- Menção ao objectivo do relatório ("Relatório do Trabalho de Projecto")
- Opcionalmente, uma imagem alusiva ao tema do projecto
- Agradecimentos – é de bom-tom e ético agradecer a todas as entidades que contribuíram para a realização da investigação (ex.: funcionários do centro de saúde, serviço de...).
- Sumário (vulgarmente chama-se índice) – indica as principais divisões do trabalho e respectivos capítulos, assim como a respectiva paginação.
- Lista de abreviaturas/Símbolos/Sinais/Acrónimos – é um elemento recomendável à boa compreensão do relatório.

2.2. Parte textual

A introdução do relatório deve:

- justificar a escolha do tema;
- enquadrar o tema no meio da escola ou da turma;
- apresentar os objectivos do trabalho incluindo as questões surgidas;
- definir a metodologia adoptada;
- apresentar o plano de desenvolvimento do trabalho.

O corpo do relatório (desenvolvimento) relata as várias fases do trabalho desenvolvido, sendo importante destacar:

- quem fez o quê e como (descrição e justificação dos procedimentos utilizados);
- descrição do processo de investigação (incluindo tabelas e/ou esquemas, esboços de gráficos, organização dos dados recolhidos...), das pesquisas elaboradas, quando e onde e dificuldades encontradas;
- competências específicas adquiridas no âmbito das ferramentas necessárias para o desenvolvimento do projecto;
- ferramentas (informáticas e outras) utilizadas;
- cumprimento das metas intermédias e dos prazos;
- dificuldades encontradas e as formas utilizadas para as superar.

O ponto de partida para a elaboração do desenvolvimento do relatório devem ser as anotações feitas em cada aula, a planificação das tarefas e o cronograma.

No caso do relatório final, o corpo do relatório deve incluir também a caracterização do produto final do projecto, que pode ser uma descrição ou um manual de utilização do que foi feito.

Por fim, as conclusões e os comentários finais devem traduzir os sentimentos dos alunos em relação ao que foi feito ao longo do ano, assim como observações críticas e sugestões de trabalhos futuros. Pode ainda incluir uma apreciação autocritica sobre a intervenção de cada elemento do grupo.

2.3. Parte referencial

Anexos

- Nos anexos, devem constar os diagramas, imagens e documentação, que iriam sobrecarregar a secção de desenvolvimento.
- Permitem fornecer informações mais completas, englobando documentos auxiliares necessários à elaboração do trabalho de projecto (ex.: legislação consultada, entrevistas realizadas...).
- A paginação deve continuar a do texto principal e, sempre que seja necessário, apresentar diferentes anexos que devem ser identificados por uma letra maiúscula (ex.: Anexo A – Legislação na área do xxxx; Anexo B – Entrevistas realizadas aos responsáveis do Serviço xxxx).
- As figuras devem ser numeradas com algarismos árabes e a legenda e fonte devem ser escritas por baixo da figura correspondente. As fotografias são tratadas como figuras.
- As tabelas devem ser numeradas com numeração romana, com as legendas escritas por cima da tabela correspondente e a fonte por baixo.

Actividade n.º 9 - Relatório final

objectivos:
neste relatório pretende-se que faças uma reflexão crítica sobre as diversas fases do desenvolvimento do teu projecto e ainda sobre o teu empenho e atitude cívica.

visão de conteúdos:

apresentam-se alguns pontos que funcionam como guias para a elaboração do relatório, sem prejuízo de outros aspectos que consideres importantes e também queiras referir.

análise fundamentada dos contributos do trabalho de projecto, para:

imobilização e integração das competências adquiridas nas diferentes disciplinas;

capacidade de trabalho autónomo e em equipa;

recolha e organização de informação;

utilização de ferramentas simples de tratamento de informação.

avaliação da concepção e planificação do projecto, tendo em conta os constrangimentos surgidos das estratégias para conseguir os resultados obtidos e os produtos elaborados.

avaliação do desenvolvimento do projecto, tendo em conta o projecto inicial.

avaliação do impacto do projecto na comunidade escolar/na turma/nos elementos do grupo de trabalho.

análise das vantagens e dificuldades do trabalho colaborativo, reflectindo sobre os problemas concretos surgidos no grupo e referindo o seu empenho pessoal.

reflexão sobre as mudanças (atitudes e competências) que o trabalho de projecto operou no teu caso concreto.

avaliação do que farias de diferente se iniciasses agora o projecto.

Observação: Consulta a pág. 131: Apoio 2 – Como redigir um relatório.



Actividades

Actividade n.º 7 – Ajuda intergrupos

Sumário da actividade:

Quando os grupos estão a trabalhar nos seus projectos, é útil ter estratégias para os grupos partilharem problemas entre eles. Os alunos aprendem entre si, geram ideias novas e obtêm feedback dos outros grupos.

Materiais:

- Apontamentos de investigação
- Projecto até ao momento

O que fazer?

- Um modo de partilhar informação é ter um grande *placard* na sala onde os alunos possam colocar as suas perguntas/problemas. Os outros grupos podem registar respostas no mesmo. Também podemos usar as perguntas para gerar uma discussão com a turma toda acerca de uma questão que seja de interesse geral.
- Outro modo de partilhar a informação é a mesa-redonda informativa. Os grupos usam os seus apontamentos para seleccionar questões que gostariam de ver respondidas e formulam essas questões à turma. Cada grupo pode pôr um par de questões.
- Tendo computadores na sala e se são usados com regularidade, podem colocar as questões electrónicamente por e-mail, numa base de dados, etc.

Actividade n.º 8 – Relatório intermédio

Com este relatório pretende-se que faças uma reflexão sobre todo o processo de desenvolvimento do projecto até ao momento. Assim, apresentam-se alguns pontos que funcionam como guias para estruturar o teu relatório.

Introdução:

- Contextualização do trabalho individual no grupo.

Desenvolvimento:

- Descrição das actividades realizadas (nas várias fases do trabalho), do processo de investigação, das pesquisas elaboradas, quando, onde e sua justificação.
- Análise das dificuldades sentidas e dos modos de superação. Reflexão sobre o cumprimento das metas e dos prazos. Ponderação sobre os desafios, as mudanças de atitude e competências exigidas pelo desenvolvimento do projecto.

Avaliação:

- Auto-avaliação (avaliação do trabalho desenvolvido individualmente e pelo grupo, devidamente justificada).

Sugestão: Consulta a pág. 131: Apoio 2 – Como redigir um relatório.

Actividades

Actividade n.º 5 – Ficha de análise de um site

Análise de documento obtido através de pesquisa na Internet.

Aluno:

N.º: _____ Ano/Turma: _____

Data: 20 _____

Site	
Organismo responsável pelo site	
Título do documento	
Data da pesquisa	
Autor (se indicado)	

Síntese do documento:

Tópicos/Assuntos do projecto com o(s) qual(is) é possível relacionar o documento:

- _____
- _____
- _____
- _____
- _____
- _____
- _____

Explicação da relação entre o documento e (pelo menos) um tópico do projecto:

atividade n.º 4 - Técnicas de investigação

cenário da actividade:

nos de escolhidos os temas, os grupos podem ter dificuldade em começar a investigar. Esta actividade ajuda os alunos a preparam um plano de investigação do seu tema usando discussões/brainstorming em grupo e relatórios de progresso (orais/escritos). O professor e os grupos terão oportunidade de rever os planos de investigação, proporcionando um valioso feedback aos alunos. Professor e alunos devem ter oportunidade de planos anteriores para comparar com os planos reformulados ou o plano final para rever e avaliar os progressos.

Materiais:
Material de escrita e papel (para a sessão de brainstorming)

Actividade de brainstorming (ver pág. 120)

Acesso a materiais de pesquisa, tais como: livros, jornais, revistas, vídeos, Internet, etc.

Acesso a recursos de profissionais, como peritos, grupos comunitários, bibliotecários, etc.

que fazer?

1 Esta actividade deve começar com uma discussão ao nível da turma acerca do que é uma investigação. Sugerem-se algumas das questões a apresentar aos alunos:

- O que é a investigação? O que é um plano de investigação? O que é que significa e o que é que implica a investigação nas diferentes profissões? Que actividades de investigação é que os alunos já desenvolveram?

- Quais os recursos disponíveis para os alunos na escola e na comunidade? Em que recursos adicionais podem pensar?

- Quais são os requisitos de investigação que precisam ter para este projecto? O que é uma fonte primária? De quantas fontes precisam? Vão entrevistar alguém? Vão elaborar inquéritos?

2 Seguidamente, os alunos em brainstorming fazem uma lista de ideias de investigação. Nesta sessão podem trabalhar por grupos de projectos individuais ou como turma. São encorajados a formular outras perspectivas de abordagem do problema e devem ter tempo para apresentar cada projecto.

3 No período de discussão a seguir à sessão de brainstorming, os alunos ou os grupos criam planos escritos para orientarem a sua investigação. Embora estes planos se possam alterar à medida que aprendem mais acerca dos assuntos e que se familiarizam com os recursos disponíveis, este plano ajuda a começar e serve de guião. As seguintes perguntas fazem parte da discussão:

- Estamos a considerar mais de uma perspectiva acerca do tema?

- Quais são as vantagens das fontes primárias e das secundárias?

- Estamos a considerar demasiadas fontes?

- Consideramos as fontes na Internet, em livros, em jornais ou outras?

- As nossas fontes são de confiança?

- Estamos a afastar-nos do nosso tema?

- Estamos a levantar as questões que interessam?

- Que trabalho podemos fazer em cada dia/semana?

- Quando é que sabemos que temos informação suficiente?

Os grupos e o professor podem rever ou criticar os planos de investigação usando as questões atrás e juntando quaisquer sugestões ou pistas para oportunidades de investigação. Os alunos/grupos devem sentir-se livres para restringir ou redefinir os temas à medida que progridem.

Variações:

Existem vários outros formatos para os alunos apresentarem os resultados das suas pesquisas ou o seu projecto final. Antes ou depois de completarem os seus projectos, os alunos deverão preparar e apresentar a forma de um trabalho escrito, uma apresentação oral, um póster, etc.

Actividades

Actividade n.º 3 – Planificação do projecto

12.ª Classe

Tema aglutinador:

Subtema:

1. IDENTIFICAÇÃO DO PROBLEMA

--

2. OBJECTIVOS

--

3. PRODUTO FINAL

--

4. INTERDISCIPLINARIDADE

Disciplinas intervenientes	Formas de intervenção

5. RECURSOS/MATERIAIS

--

6. CALENDARIZAÇÃO

--

7. DESENVOLVIMENTO DAS ACTIVIDADES

Grupo de trabalho	Distribuição de tarefas

Actividade n.º 2 - Relatório de escolha do tema

Objectivo:

Em este relatório pretende-se que faças uma reflexão sobre todo o processo de escolha do tema/problema do projecto.

Materiais:

• Folha de exercício (pautada), caneta azul ou preta

• Material usado até ao momento

• Documentos gerados nas reuniões para a escolha do tema

Guião de conteúdos:

Apresentam-se alguns pontos que funcionam como guias para elaboração do relatório, sem prejuízo de outros aspectos que consideres importantes e também queiras referir.

• Reflexão sobre os desafios e mudanças de atitude exigidas pelo trabalho de projecto.

• Descrição do processo de selecção de temas a investigar/abordar (dificuldades, temas abandonados, etc.).

• Relação do tema escolhido com a área vocacional e as disciplinas que frequenta actualmente.

• Indicação das questões orientadoras e das pistas de investigação, especificando a sua autoria.

• Análise da influência da constituição da equipa de trabalho na escolha do tema.

• Avaliação das potencialidades do tema escolhido: produtos possíveis, impacto esperado.

Especificações:

• Respeito pelas margens, aspecto limpo e cuidado, poucas emendas e rasuras.

• Titulação e secções bem definidas.

• Conteúdos organizados e bem estruturados (ver pág. 131: Apoio 2 – Como redigir um relatório).

4.4. Investigar e produzir

O contacto com o meio através do trabalho de campo é o momento privilegiado da recolha de dados que depois terão de ser tratados. Nesta fase, os alunos realizam as tarefas conforme o planeamento definido anteriormente.

Devem recorrer a diferentes técnicas e ferramentas para encontrar as respostas às questões e executar as tarefas planeadas.

A utilização da Internet é fundamental, uma vez que através desta podem realizar pesquisas, consultar especialistas sobre o tema em estudo, enviar e receber questionários e aceder a um conjunto de recursos que podem auxiliar o aluno na execução das tarefas.



Fig. 5

Os alunos devem ser responsáveis pelas suas tarefas e, sempre que estas não estejam a ser realizadas de acordo com o planeado, todos os elementos do grupo devem tomar conhecimento dessa situação e, em conjunto, definir acções que permitem ultrapassar as dificuldades encontradas e manter a execução do projecto o mais aproximada possível do plano previamente realizado.

No decorrer do trabalho de campo, os alunos devem registar todas as actividades e acções realizadas, pessoas contactadas, métodos de trabalho, técnicas utilizadas para a escolha de informação, reflexões e análises realizadas e eventualmente conclusões.

No decorrer do trabalho de campo é importante que o grupo de trabalho faça o ponto da situação (momento do trabalho de projecto em que os alunos devem fazer uma reflexão sobre as tarefas realizadas, a realizar e a sua calendarização) com alguma frequência. Ao fazê-lo, o grupo tem capacidade para analisar o trabalho realizado e confrontá-lo com o planeamento.

Consulta o Apoio 3 da página 133:
Técnicas de investigação

Realiza a Actividade n.º 4 da página 123:
Técnicas de investigação

Consulta o Apoio 4 da página 135:
Pesquisar na Web

Realiza a Actividade n.º 5 da página 124:
Ficha de análise de um site

Consulta os Apoios 5 e 6 das páginas 136 e 137:
Como fazer um inquérito
Como realizar uma entrevista

São definidas as tarefas a levar a cabo pelos diferentes elementos do grupo, a escolha dos métodos e técnicas de pesquisa e a respectiva calendarização.

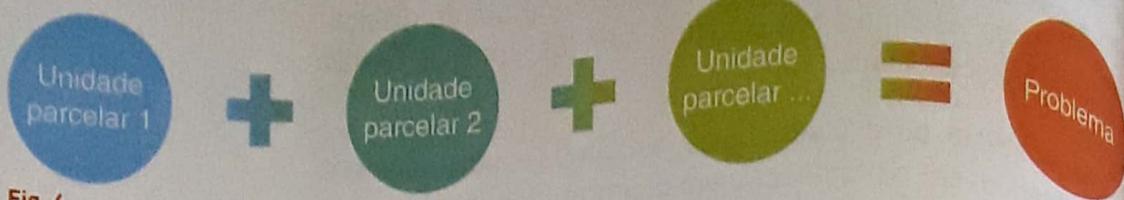


Fig. 4

4.3. Esboçar e planificar

Nesta fase, os alunos devem preparar o trabalho, elaborando um plano de actividades e tarefas a realizar durante a execução do projecto. No início do projecto, o conhecimento do mesmo na totalidade é normalmente muito vago, pelo que existe necessidade de, ao longo da sua execução, rever com alguma frequência o planeamento, de forma a identificar eventuais atrasos na execução das tarefas e desencadear as acções necessárias para a correcção dos desvios verificados.

Como resultado desta fase, os alunos devem elaborar um documento que descreva as várias fases do projecto e respectivas tarefas, enquadramento temporal, recursos necessários para a sua realização e o responsável ou responsáveis pela sua execução. Com a colaboração dos professores envolvidos no projecto, devem ser ainda identificadas as situações mais complexas que possam de alguma forma dificultar o projecto.

Apresentamos de seguida um resumo dos itens a incluir no documento com a planificação do trabalho a realizar:

Preenche o documento da Actividade n.º 3 da página 122:
Planificação do projecto

- identificação dos objectivos do projecto;
- definição de tarefas e actividades;
- identificação dos recursos necessários para a realização das tarefas;
- divisão de tarefas e responsabilidades;
- calendarização de tarefas.

Consulta o Apoio 9 da página 143:
O cronograma

Nesta fase, o recurso a ferramentas e aplicações abordadas nesta disciplina, na 11.ª classe, é inevitável e obviamente desejável. Os alunos devem elaborar um **cronograma** (apresentação e discriminação das várias tarefas a realizar, tempos de execução e datas previstas de concretização), o qual pode ser criado utilizando o processador de texto. Esta aplicação também pode ser utilizada para elaborar o documento com a planificação de todo o trabalho.

1. CONCEITO

O Trabalho de Projecto é um método de trabalho que se centra na identificação, investigação, análise e resolução de problemas em grupo. O termo "projeto" significa "lançar para a frente, atirar". Projectar significa investigar um tema, um problema, uma situação com o objectivo de a conhecer e, se possível, apresentar interpretações e/ou soluções novas.

Uma das características mais marcantes do Trabalho de Projecto é o papel do aluno no processo de aprendizagem; o trabalho está centrado nos alunos porque são eles que escolhem os temas, os problemas dos projectos que vão desenvolver, investigar e apresentar como produto final.

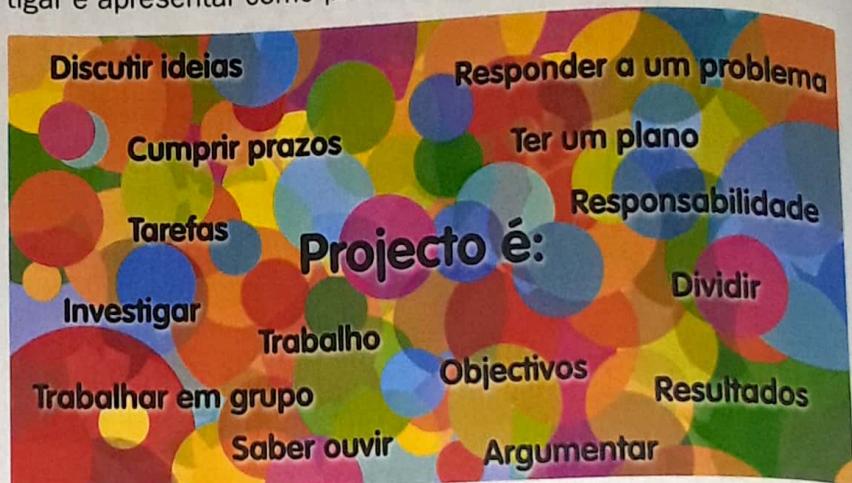


Fig. 1

2. PAPEL DOS PROFESSORES

Na escolha do tema, cabe ao professor a tarefa de, dentro do possível, uniformizar a dificuldade dos trabalhos propostos pelos alunos. Quanto ao número de temas por turma, depende das possibilidades do professor, do número de alunos por turma e das características do grupo de trabalho.

O professor deve procurar acompanhar o trabalho dos alunos no sentido de os alertar para eventuais dificuldades e ajudá-los a ultrapassá-las. Deve incentivá-los à discussão de ideias e troca de experiências vividas na execução das tarefas, para que todos possam aprender com as experiências dos outros.

Os professores devem fazer uma síntese do trabalho realizado, analisando os seguintes aspectos:

- métodos de trabalho, dificuldades e a forma como estes foram ultrapassados;

Actividade n.º 1

1 Por que razão deves instalar uma ferramenta antivírus no teu computador?

2 Como podes obter uma ferramenta antivírus?

3 Qual a necessidade de manter a ferramenta antivírus actualizada?

4 Quais os procedimentos a seguir para garantir que a ferramenta antivírus está actualizada?

5 Quais os procedimentos a seguir para substituir uma ferramenta antivírus, já instalada no computador, por outra?

6 Refere dois sinais indicativos de que o computador tem *spyware*.

7 Indica duas formas de um computador poder ser vítima de *spyware*.

8 Como actuam os programas de *spyware*?

9 Como actuar no caso de o computador ter *spyware*?

10 Pretendes realizar uma palestra para alertares os teus colegas mais jovens da necessidade de terem instalado no computador um antivírus.

Para tal:

- recorrendo ao teu livro e à Internet, recolhe a informação necessária sobre o tema, nomeadamente, sobre os perigos dos vírus informáticos e como os prevenir;
- cria um cartaz apelativo para expores na escola;
- cria uma apresentação em PowerPoint que possa ser visionada pelos teus colegas.

11 Pretendes realizar uma palestra para alertares os teus colegas mais jovens da necessidade de terem instalada no computador uma ferramenta *antispyware*.

Para tal:

- recorrendo ao teu livro e à Internet, recolhe a informação necessária sobre o tema, nomeadamente sobre como actuam os *spyware* e como os evitar;
- cria um cartaz apelativo para expores na escola;
- cria uma apresentação em PowerPoint que possa ser visionada pelos teus colegas.

Navegar e fazer transferências de ficheiros com segurança

O ideal seria não efectuar qualquer tipo de transferência de informação entre o computador e a Internet. Mas sabemos que isto é impossível. Assim, há alguns aspectos a que deves dar atenção, nomeadamente:

- transferir ficheiros apenas de sites em que confies (fidedignos);
- ao instalar um software transferido da Internet, lê cuidadosamente todos os acordos, o contrato de licença e a declaração de privacidade, pois, muitas das vezes, a inclusão de software indesejado está documentada, mas pode ser apresentada entre outro texto sem nos apercebermos;
- lê atentamente toda a informação que apareça e não faças qualquer tipo de confirmação sem ter a certeza do que se trata.

15.2.2. Eliminar *spyware*

Tal como os vírus, também o *spyware* pode ser difícil de eliminar do nosso computador, pois, normalmente, estes programas, mesmo depois de desinstalados pelo método normal, reaparecem quando se reinicia o computador.

Para remover o *spyware*:

- transfere e instala o **Windows Defender** ou qualquer outra ferramenta do tipo que tenhas decidido adoptar;
- executa o programa para fazer a pesquisa de ficheiros.

Selecciona os ficheiros que devam ser removidos e segue as instruções do programa.



Fig. 18

Actividades

Actividade n.º 1

- 1** Identifica o tipo de cada uma das seguintes constantes:
- a. 512
 - b. $-12,012 \cdot 10^{13}$
 - c. 134,2
 - d. $27,4 \cdot 10 - 33$
 - e. "745"
 - f. "TERMINAR"
- 2** Indica o resultado e o tipo de cada uma das expressões:
- a. $5 + 3^2 \cdot 2$
 - b. $4^2 - 10 + (2^3 - 5)$
 - c. $1/4 + 2$
 - d. $7^2 - 4 \cdot 2 + 3$
 - e. $29,0/7 + 5$
 - f. $40 - [5^2 + (2^3 - 7)]$
- 3** O valor da expressão $(-4)^4 - (-4)$ é:
- a. 160
 - b. 260
 - c. -160
 - d. -260
- 4** Quais dos seguintes são considerados nomes inválidos de variáveis? Justifica.
- a. XX
 - b. X/Y
 - c. 1A
 - d. TEXTO
 - e. A3.1
 - f. MOD
- 5** Indica o valor da variável FINAL depois da execução das instruções.
- a. $FINAL \leftarrow 5 + 3^2 \cdot 2$
 - b. $FINAL \leftarrow 14$
 - c. $A \leftarrow 2$
 - d. $VALOR \leftarrow 16$
 - e. $B \leftarrow 3$
 - f. $FINAL \leftarrow (FINAL + VALOR) / 2$
 - FINAL $\leftarrow A^B - B$
- 6** Escreve as seguintes expressões matemáticas como expressões de computação:
- a. $\frac{a}{b} + 1$
 - b. $a + \frac{b}{c - d}$
 - c. $[(a + b)^c]^d$
 - d.
$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$
- Para cada um dos exercícios seguintes, apresenta o algoritmo em linguagem natural através de um fluxograma e em pseudocódigo.
- 7** Pretende-se imprimir todos os números inteiros entre 2 e 200.
Cria o algoritmo em linguagem natural, desenha o respectivo fluxograma e cria o algoritmo em pseudocódigo.
- 8** Pretende-se determinar a soma dos números positivos inferiores ou iguais a 1000.
Cria o algoritmo em linguagem natural, desenha o respectivo fluxograma e cria o algoritmo em pseudocódigo.
- 9** Pretende-se determinar se um determinado número, dado pelo utilizador, é par ou ímpar, positivo ou negativo.
Cria o algoritmo em linguagem natural, desenha o respectivo fluxograma, cria o algoritmo em pseudocódigo e faz uma traçagem considerando o valor - 22.

Passo	SOMA	CICLO	NOME	N_ALUNO	MEDIA	NOTA[CICLO] > MEDIA	Resultado
13	56						
14		5					
15							
16	70		Luís	14			
17					14		
18		1					
19						NOTA[1] = 12 12 > 14 Falso	
20		2					
21						NOTA[2] = 8 8 > 14 Falso	
22		3					
23						NOTA[3] = 16 16 > 14 Verdadeiro	Maria 16
24		4					
25						NOTA[4] = 20 20 > 14 Verdadeiro	Augusto 20
26		5					
27						NOTA[5] = 14 14 > 14 Falso	

O algoritmo a seguir apresentado é uma versão simplificada do anterior. O resultado é exactamente o mesmo, mas os nomes e as notas dos alunos são lidos directamente para o vector e não para as variáveis. Analisa-o cuidadosamente.

Algoritmo NOTA_MED_V2: este algoritmo determina a média das notas de cinco alunos e imprime o nome e a nota dos alunos que se posicionem acima da média (nota **superior** à média). O nome e a nota de cada aluno devem ser introduzidos pelo utilizador.

```

SOMA ← 0
Repete para CICLO = 1 até 5
  Ler (ALUNO[CICLO], NOTA[CICLO])
  SOMA ← SOMA + NOTA[CICLO]
Fim para
MÉDIA ← SOMA / 5
Repete para CICLO = 1 até 5
  SE NOTA[CICLO] > MÉDIA
    Então Escrever (NOME[CICLO], NOTA[CICLO])
Fim para
Fim

```

cujo fluxograma pode ser representado da seguinte forma:

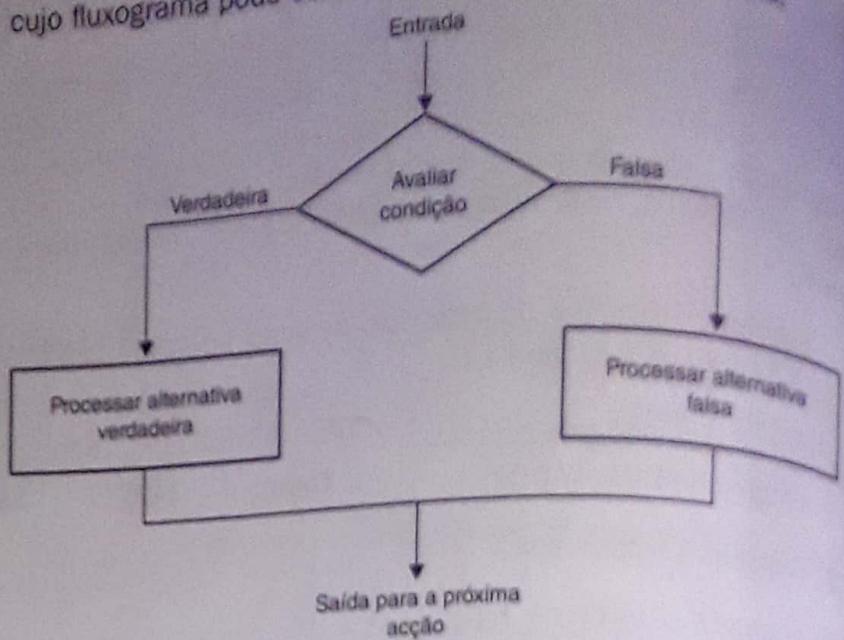


Fig. 6

Exemplo: Supõe que N1 e N2 são variáveis numéricas diferentes e pretendemos imprimir a maior.

Assim, se o valor de N1 for maior que o valor de N2 pretendemos que o valor da variável N1 seja impresso. Se o valor de N2 for maior que o valor de N1 pretendemos que seja impresso o valor da variável N2.

Temos de determinar se N1 é maior que N2, utilizando a expressão **N1 > N2** que descreve a condição a testar.

Este teste é feito durante a execução do programa e, dependendo do resultado, será impresso um ou outro valor.

Fluxograma:

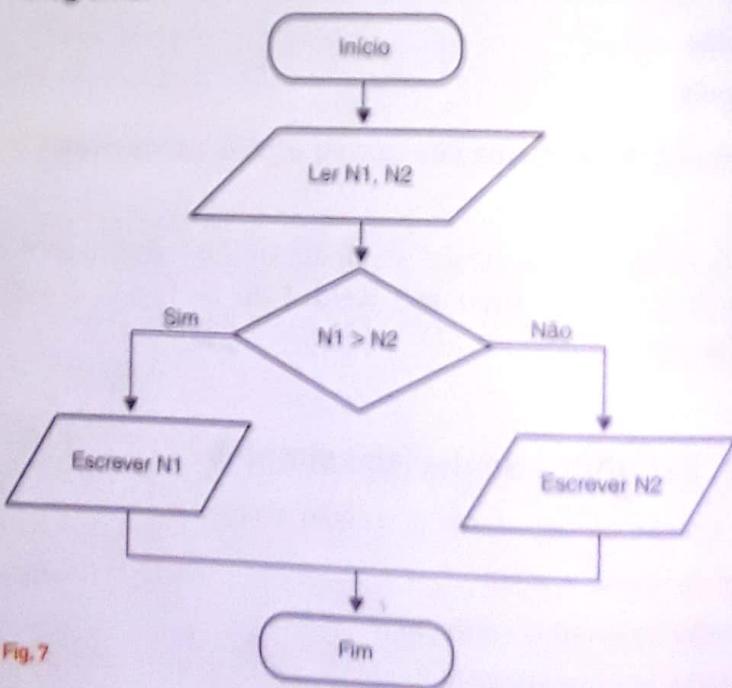


Fig. 7

9.4. Traçagem de algoritmos

Depois de desenvolvido o algoritmo é importante determinar se ele, efectivamente, faz o que é suposto fazer. Para tal, o algoritmo é executado manualmente, com dados representativos, e registados todos os valores que as variáveis assumem em cada um dos passos. Esta técnica é conhecida por *Tracing* (traçagem).

Exemplo: Traçagem do algoritmo descrito no ponto anterior (9.3.) com os valores 4 e 3.

1. Ler os lados do triângulo retângulo (LADO1, LADO2).
2. Calcular a hipotenusa.
3. Escrever o valor da hipotenusa.

Passo	LADO1	LADO2	HIPOTENUSA	Resultado
1	4	3		
2	4	3	5	
3	4	3	5	5

10. ESTRUTURA DE DECISÃO

Normalmente, a resolução de um problema exige que o programa, durante a sua execução, seja capaz de tomar decisões, podendo, consequentemente, alterar o seu curso de acção.

A componente lógica permite ao computador tomar decisões.

A base da estrutura é:

Se

Então

Senão

e permitirá ao programa seleccionar acções alternativas.

A decisão é especificada numa expressão lógica e, conforme o resultado desta expressão (verdadeiro ou falso), é seguida uma acção.

10.1. Estrutura de decisão simples

A sintaxe da estrutura de decisão simples é:

Se condição a testar

Então alternativa verdadeira

Senão alternativa falsa

A condição é testada. Se for verdadeira, é executada a alternativa verdadeira. Senão, é executada a outra alternativa (alternativa falsa).

9.3. Linguagem algorítmica (pseudocódigo)

A combinação da abordagem narrativa e do fluxograma resulta num tipo de linguagem de programação, simplificada, a que chamamos **pseudocódigo**.

Apesar de se pretender que não seja muito próxima de nenhuma linguagem de programação, na realidade acaba sempre por haver alguma influência da linguagem a que estamos mais habituados. Importa, portanto, que seja suficientemente geral para que a passagem do algoritmo para o programa seja o mais directa possível, independentemente da linguagem de programação adoptada.

Exemplo: Pseudocódigo correspondente ao algoritmo descrito no ponto 9.1..

Algoritmo Hipotenusa: Dados os lados de um triângulo rectângulo, determinar o valor da hipotenusa.

A palavra **Fim** indica que o algoritmo termina.

Ler (LADO1, LADO2)

HIPOTENUSA \leftarrow SQRT(LADO1 2 + LADO2 2)

Escrever (HIPOTENUSA)

Fim

Repara que ao algoritmo foi atribuído um nome (Hipotenusa) e foi feita uma breve descrição da sua função. Em seguida foram apresentados os comandos que executam as operações necessárias.

Se pretendêssemos, poderíamos ter documentado cada um dos passos ou, pelo menos, os mais importantes.

Exemplo:

Algoritmo Hipotenusa: Dados os lados de um triângulo rectângulo, determinar o valor da hipotenusa.

[Ler os lados do triângulo rectângulo]

Ler (LADO1, LADO2)

[Calcular a hipotenusa]

HIPOTENUSA \leftarrow SQRT(LADO1 2 + LADO2 2)

[Imprimir o valor da hipotenusa]

Escrever (HIPOTENUSA)

Fim

Este rigor não é obrigatório, pois, em situações um pouco mais complexas, pode tornar o algoritmo demasiado confuso.

Forma	Operação que representa	Nome
oval	Ínicio e final	Terminador
retângulo	Cálculo	Rectângulo
losango	Decisão	Losango
paralelogramo	Entrada e saída de dados	Paralelogramo

As formas são ligadas por setas que indicam a ordem e o sentido em que as operações devem ser executadas.

Exemplo: Fluxograma correspondente ao algoritmo descrito no ponto anterior.

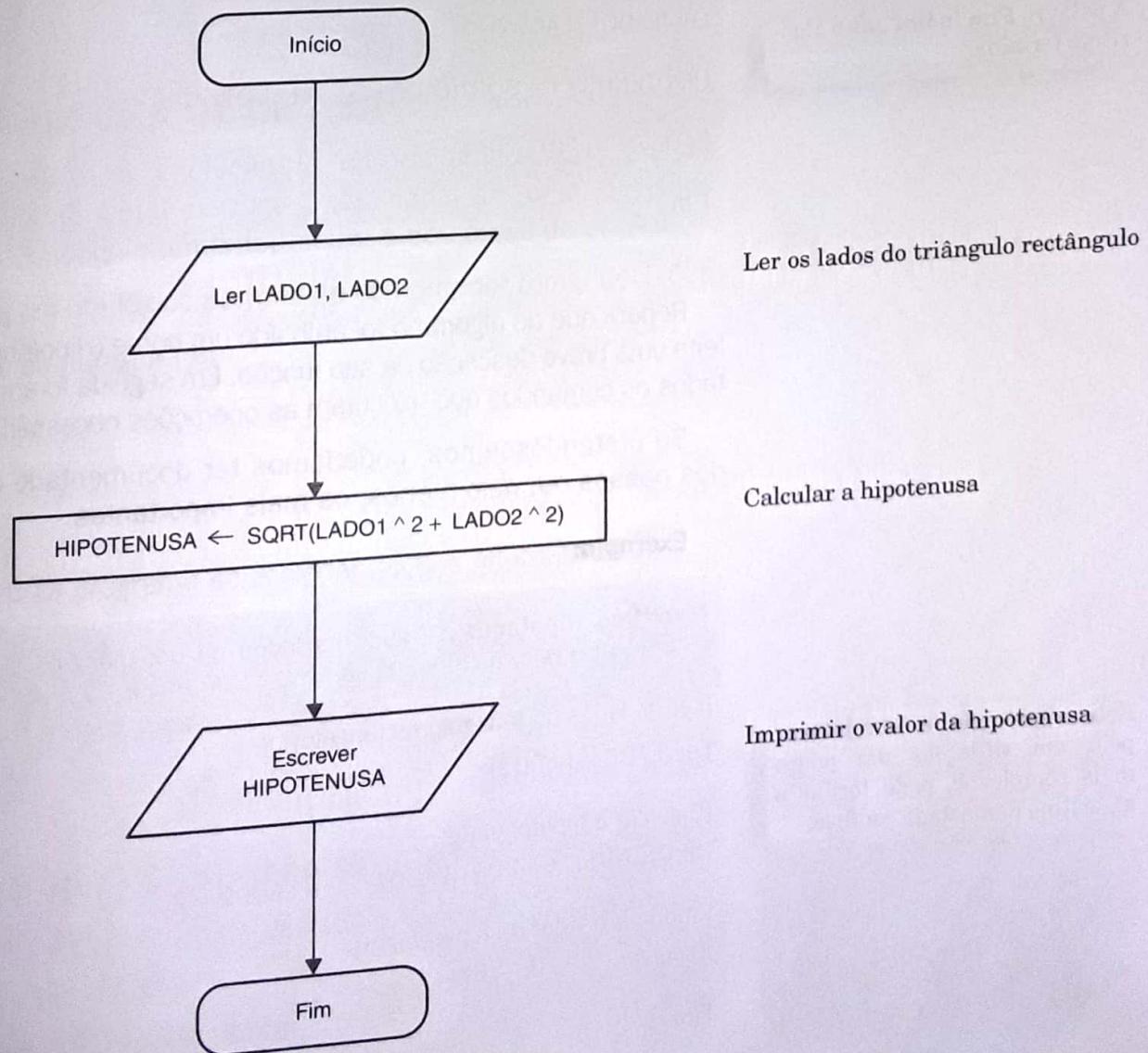


Fig. 5

9. DESCRIÇÃO DE ALGORITMOS

A criação de um algoritmo deve ser o ponto de partida para o desenvolvimento de um programa informático. Como já foi referido, cada passo de um algoritmo deve ser o mais simples possível e sem ambiguidades. Assim, a sua posterior transformação em programa será facilitada, bem como a elaboração da documentação necessária.

9.1. Descrição natural

As receitas culinárias são, normalmente, descritas desta forma.

Este é um método directo de expressar um algoritmo e consiste em especificar em linguagem corrente cada um dos seus passos.

Contudo, é muito difícil transmitirmos, com precisão, algumas situações e condições que o algoritmo deve contemplar. Efectivamente, a imprecisão da linguagem natural pode induzir a uma má interpretação e a perda de informação.

Por estes motivos, a linguagem natural não deve ser utilizada como única ferramenta para expressão de algoritmos, sendo, contudo, muito útil para documentar, esclarecendo ou realçando situações especiais.

Exemplo:

Cria um algoritmo, em linguagem natural, que, dados os lados de um triângulo rectângulo, determine a hipotenusa.

Algoritmo Hipotenusa

1. Ler os lados do triângulo rectângulo
2. Calcular a hipotenusa
3. Imprimir o valor da hipotenusa

9.2. Diagramas lógicos – fluxogramas

Um **fluxograma** é um diagrama lógico que permite descrever um algoritmo, reflectindo a sua lógica e realçando cada um dos passos e as suas interconexões.

Num fluxograma são utilizadas as seguintes formas:

8.1. Comando de entrada

O comando de entrada de dados (leitura) que iremos utilizar tem a seguinte sintaxe:

Ler (lista de entrada)

A lista de entrada é constituída pelas variáveis às quais os valores devem ser atribuídos, na mesma ordem em que estão especificados.

Ler (A, B, C)

Este comando faz com que sejam lidos três valores e atribuídos às variáveis na mesma ordem. Assim, o primeiro valor lido é atribuído à variável A, o segundo à variável B e o terceiro à variável C.

8.2. Comando de saída

O comando de saída de informação (escrita) que iremos utilizar tem a seguinte sintaxe:

Escrever (lista de saída)

A lista de saída é constituída pelas variáveis cujo conteúdo pretendas conhecer. Assim, é possível mostrar o conteúdo de qualquer variável, o resultado de uma expressão ou o valor de uma constante.

Claro que nesta fase não temos a preocupação do formato da saída da informação, assumindo que este é o mais adequado.

A ← 10,0
B ← 14,5
C ← 16,0
MEDIA ← (A + B + C) / 3

Escrever (MEDIA)

Como resultado desta sequência de comandos, no ecrã seria visualizado o número 13,5.

Se, para além da média dos três valores, também quiséssemos visualizar cada um dos valores que constituíram o cálculo, o comando de saída seria:

Escrever (A, B, C, MEDIA)

Se pretendêssemos que antes do valor da média fosse impressa a mensagem **A média dos valores é:**, o comando adequado seria:

Escrever ("A média dos valores é:", MEDIA)

Repara que o texto é colocado entre aspas. Contudo, as aspas não serão visualizadas.

7.2. Funções embutidas

Os operadores convencionais (+, -, *, /) não nos permitem especificar todas as operações requeridas, sendo, vulgarmente, complementados por operadores especiais, denominados funções embutidas.

Assim, podemos dizer que funções embutidas são programas já incluídos nas linguagens de programação, para auxiliar o programador na execução de cálculos.

Algumas das funções embutidas mais comuns, e que são fornecidas na maioria das linguagens de programação, são as seguintes:

Nome da função	Significado	Argumento	Tipo do resultado
ABS()	Valor absoluto	Expressão real ou inteira	Mesmo tipo do argumento
SQRT()	Raiz quadrada	Expressão real ou inteira ≥ 0	Real
TRUNC()	Valor truncado	Expressão real	Real
ROUND()	Valor arredondado	Expressão real	Real
LOG()	Logaritmo base e	Expressão real	Real
LOG 10()	Logaritmo base 10	Expressão real	Real
EXP()	Exponencial	Expressão real	Real
SIN()	Seno	Expressão real	Real (em radianos)
COS()	Co-seno	Expressão real	Real (em radianos)
TAN()	Tangente	Expressão real	Real (em radianos)

Exemplo: Para calcular a hipotenusa de um triângulo rectângulo, podes utilizar a expressão:

HIPOTENUSA \leftarrow SQRT(LADO1 2 + LADO2 2)

Contudo, para que não seja produzido um erro, às variáveis LADO1 e LADO2 devem ser atribuídos valores antes da execução da expressão.

8. COMANDOS DE ENTRADA E SAÍDA

A grande importância do computador prende-se com o processamento efectuado sobre os dados que lhe fornecemos, no sentido de obter resultados nos formatos mais adequados.

As instruções de entrada e saída dependem da linguagem de programação utilizada e, muitas vezes, do próprio sistema informático. Assim, iremos aqui definir dois comandos para dar a indicação de entrada e saída.

Comando	Descrição
Ler	Comando que permite ler valores, dados pelo utilizador ou obtidos a partir de qualquer outro tipo de dispositivo de entrada, atribuindo-os às variáveis indicadas.
Escrever	Comando que permite mostrar os resultados no ecrã ou impressos.

6 Define a chave primária de cada uma das tabelas.

Nas tabelas **Alunos**, **Códigos Postais** e **Disciplinas**, o campo assumido como chave, na criação da tabela, é o correcto.

Ou seja:

Tabela	Campo-chave
Alunos	NALUNO
Códigos Postais	CPOSTAL
Disciplinas	CODDIS

Na tabela **Notas** a situação é um pouco diferente.

Pensa que:

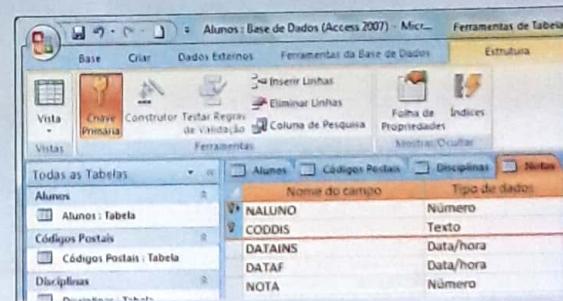
O mesmo aluno pode estar inscrito a várias disciplinas — terá várias notas — existem, na tabela, vários registo com o mesmo código de aluno — o campo **NALUNO**, sozinho, não pode ser chave.

Vários alunos fazem exame à mesma disciplina — existem, na tabela, vários registo com o mesmo código de disciplina — o campo **CODDIS**, sozinho, não pode ser chave.

Se assumires que, apesar de um aluno poder fazer mais do que uma vez exame a uma dada disciplina, apenas interessa guardar a última nota obtida, os campos **NALUNO+CODDIS** podem ser definidos como chave da tabela **Notas** (apenas existirá um registo com o mesmo código de aluno e código de disciplina).

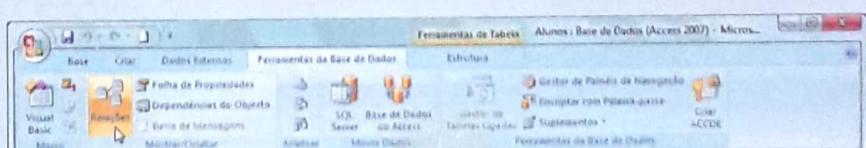
Assim:

- Seleciona os campos **NALUNO** e **CODDIS**.
- No separador **Estrutura**, clica em **Chave Primária**.
- Repara que uma **chave** aparece à esquerda dos campos seleccionados.
- Guarda a tabela.

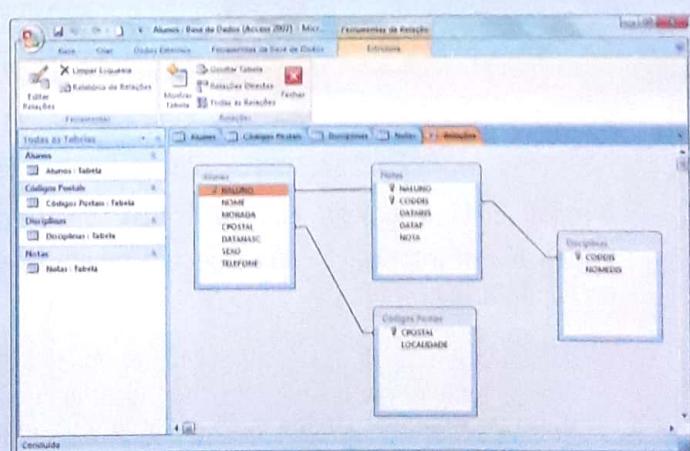


7 Estabelece as devidas relações entre as tabelas.

- No separador **Ferramentas da Base de Dados** clica em **Relações**.



- Na janela **Mostrar Tabela** selecciona as quatro tabelas e clica em **Adicionar**.
- Clica no campo e arrasta a ligação até ao campo correspondente noutra tabela.
- Guarda o esquema de **Relações**.
- Clica em **Fechar**.



Actividades

CODDIS (Código da disciplina)

Tipo de dados: Texto
 Tamanho do campo: 4
 Legenda: Código da disciplina
 Obrigatório: Sim
 Permitir comprimento zero: Não

DATAINS (Data de inscrição)

Tipo de dados: Data/Hora
 Formato: Data abreviada
 Legenda: Data de inscrição
 Regra de validação: Entre 01-01-2015 e a data actual
 Texto de validação: Data de inscrição entre 01-01-2015 e a data de hoje
 Obrigatório: Sim
 Indexado: Não

DATAF (Data de conclusão)

Tipo de dados: Data/Hora
 Formato: Data abreviada
 Legenda: Data de conclusão
 Valor predefinido: Data actual (do sistema)
 Regra de validação: Menor ou igual que a data actual
 Texto de validação: Data inválida
 Obrigatório: Sim
 Indexado: Não

NOTA

Tipo de dados: Numérico
 Tamanho do campo: Byte
 Formato: Número geral
 Casas decimais: 0
 Regra de validação: Entre 0 e 20
 Texto de validação: A nota deverá ser entre 0 e 20
 Obrigatório: Sim
 Indexado: Não

Campo: NALUNO

Propriedades do campo	
Nome do campo	Tipo de dados
NALUNO	Número
Propriedades do campo Geral Pesquisa Tamanho do campo: Número Inteiro longo Formatar: Automático Casas decimais: 0 Máscara de introdução: Número do aluno Legenda: Número do aluno Valor predefinido: Regra de validação: Texto de validação: Necessário: Sim Indexado: Sim (duplicação não autorizada) Etiquetas inteligentes: Alinhamento do texto: Geral	

Campo: CODDIS

Propriedades do campo	
Nome do campo	Tipo de dados
NALUNO	Número
CODDIS	Texto
Propriedades do campo Geral Pesquisa Tamanho do campo: 4 Formatar: Máscara de introdução: Legenda: Código da disciplina Valor predefinido: Regra de validação: Texto de validação: Necessário: Sim Permitir comprimento zero: Não Indexado: Não	

Campo: DATAINS

Propriedades do campo	
Nome do campo	Tipo de dados
NALUNO	Número
CODDIS	Texto
DATAINS	Data/hora
Propriedades do campo Geral Pesquisa Formatar: Data abreviada Máscara de introdução: Data de inscrição Legenda: Data de inscrição Valor predefinido: Entre #01-01-2015# E hoje# Regra de validação: Data de inscrição entre 01-01-2015 e a data de hoje Texto de validação: Necessário: Sim Indexado: Não	

Campo: DATAF

Propriedades do campo	
Nome do campo	Tipo de dados
CODDIS	Texto
DATAINS	Data/hora
DATAF	Data/hora
Propriedades do campo Geral Pesquisa Formatar: Data abreviada Máscara de introdução: Data de conclusão Legenda: Data de conclusão Valor predefinido: Hoje# Regra de validação: <= hoje# Texto de validação: Data inválida Necessário: Sim Indexado: Não	

Campo: NOTA

Propriedades do campo	
Nome do campo	Tipo de dados
DATAINS	Data/hora
DATAF	Data/hora
NOTA	Número
Propriedades do campo Geral Pesquisa Tamanho do campo: Byte Formatar: Número geral Casas decimais: 0 Máscara de introdução: Legenda: Nota Valor predefinido: Regra de validação: Entre 0 E 20 Texto de validação: A nota deverá ser um entre 0 e 20 Necessário: Sim Indexado: Não	

Especificações:

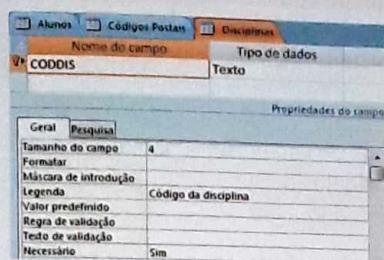
CODDIS (Código da disciplina)

Tipo de dados: Texto
 Tamanho do campo: 4
 Legenda: Código da disciplina
 Obrigatório: Sim
 Permitir comprimento zero: Não

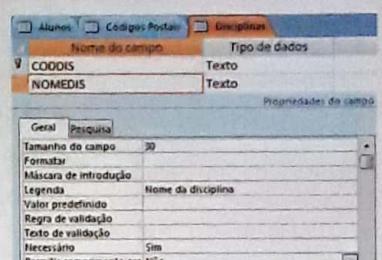
NOMEDIS (Nome da disciplina)

Tipo de dados: Texto
 Tamanho do campo: 30
 Legenda: Nome da disciplina
 Obrigatório: Sim
 Permitir comprimento zero: Não
 Indexado: Não

Campo: CODDIS



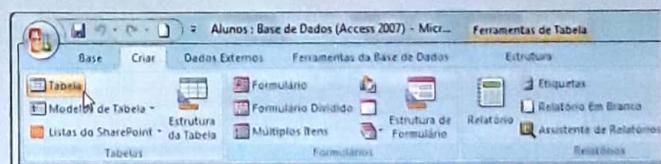
Campo: NOMEDIS



- Guarda a tabela.

5 Cria a tabela Notas tendo em atenção as especificações.

- Clica no separador Criar e selecciona Tabela.
- Clica em Vista.
- Selecciona a opção Vista de Estrutura.
- Na janela Guardar como digita Notas.
- Define cada um dos campos e compara com as figuras seguintes.



NOTAS

Número do aluno
 Código da disciplina
 Data de inscrição
 Data de conclusão
 Nota

Especificações:

NALUNO (Número do aluno)

Tipo de dados: Numérico
 Tamanho do campo: Inteiro longo
 Legenda: Número do aluno

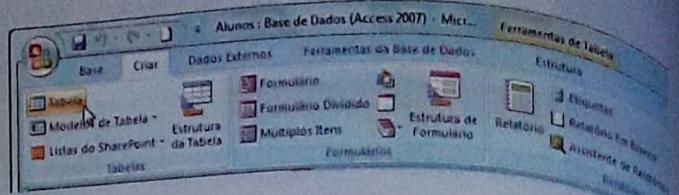
Repara que o campo **NALUNO** já não pode ser de atribuição automática. Isto porque o número do aluno, inserido na tabela **Notas**, deve existir na tabela **Alunos**.

Actividades

3

Cria a tabela **Códigos Postais** tendo em atenção as especificações.

- Clica no separador Criar e selecciona Tabela.
- Clica em Vista.
- Selecciona a opção Vista de Estrutura.
- Na janela Guardar como digita Códigos Postais.
- Define cada um dos campos e compara com as figuras seguintes.



CÓDIGOS POSTAIS

Código Postal
Localidade

Especificações:

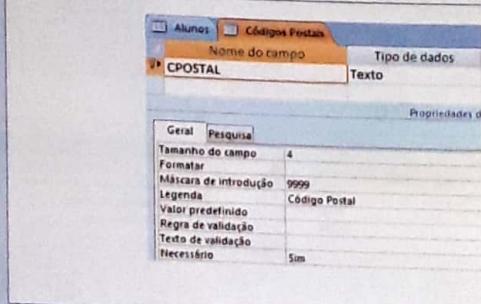
CPOSTAL (Código Postal)

Tipo de dados: Texto
Tamanho do campo: 4
Máscara de introdução: 9999
Legenda: Código Postal

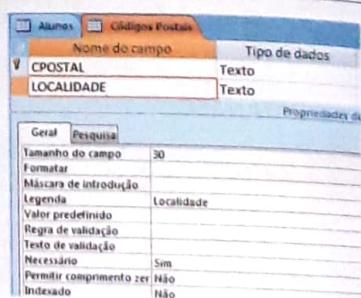
LOCALIDADE

Tipo de dados: Texto
Tamanho do campo: 30
Obrigatório: Sim
Permitir comprimento zero: Não
Legenda: Não

Campo: CPOSTAL



Campo: LOCALIDADE

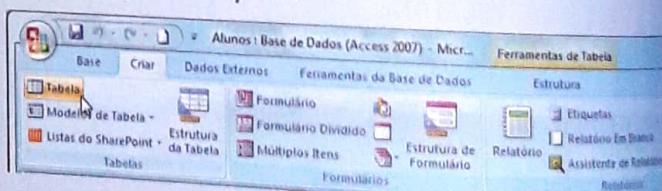


- Guarda a tabela.

4

Cria a tabela **Disciplinas** tendo em atenção as especificações.

- Clica no separador Criar e selecciona Tabela.
- Clica em Vista.
- Selecciona a opção Vista de Estrutura.
- Na janela Guardar como digita Disciplinas.
- Define cada um dos campos e compara com as figuras seguintes.

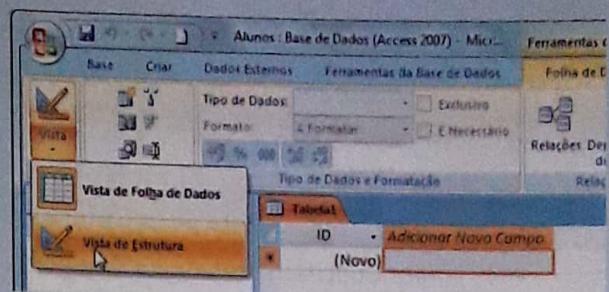


DISCIPLINAS

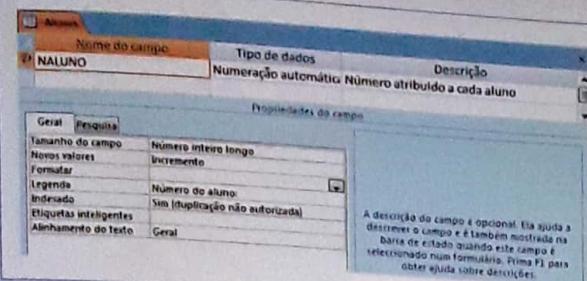
Código da disciplina
Nome da disciplina

Actividades

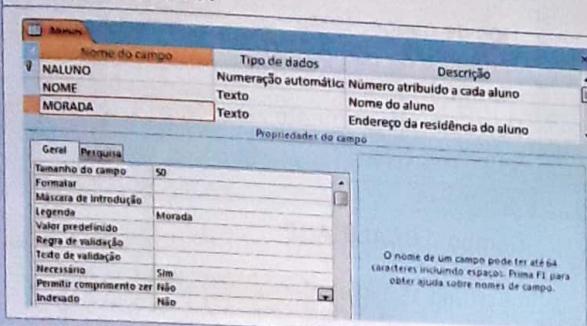
- Clica em Vista.
- Selecciona a opção Vista de Estrutura.
- Na janela Guardar como digita Alunos.
- Define cada um dos campos e compara com as figuras seguintes.



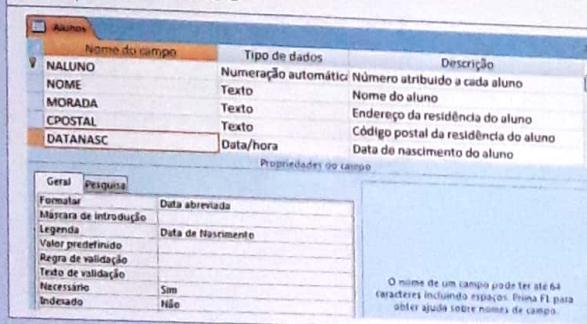
Campo: **NALUNO**



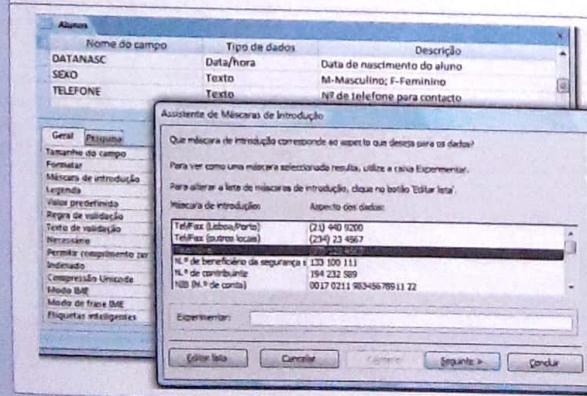
Campo: **MORADA**



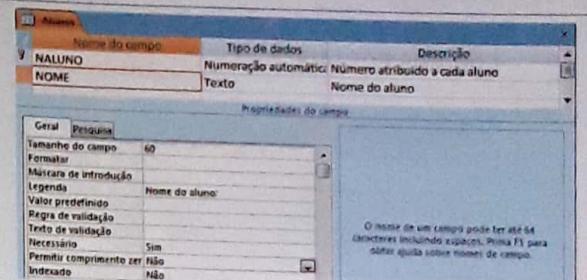
Campo: **DATANASC**



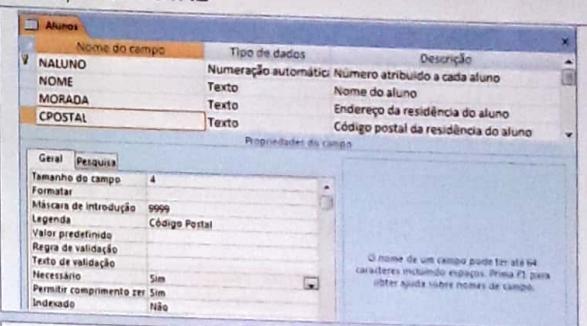
Campo: **TELEFONE**



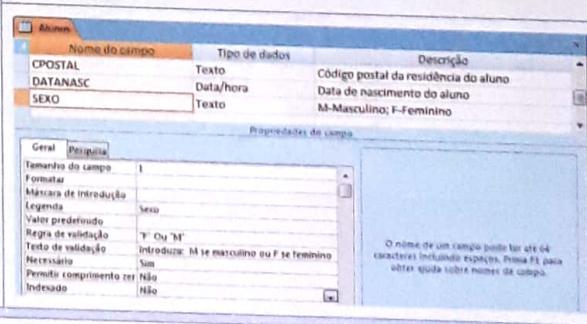
Campo: **NOME**



Campo: **CPOSTAL**



Campo: **SEXO**



- Guarda a tabela.

Actividades

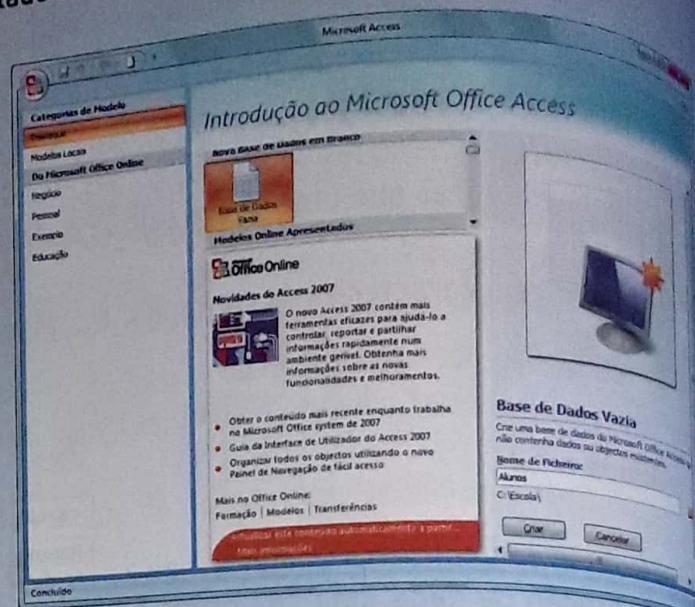
Actividade n.º 4

Efectua as seguintes tarefas e verifica se o resultado obtido é semelhante ao apresentado.

Uma escola pretende criar uma base de dados para efectuar uma gestão eficiente da informação relativa às disciplinas concluídas pelos alunos. Para tal:

1 Cria uma base de dados com o nome **Alunos**.

- Acede ao Access.
- Clica em **Base de Dados Vazia**.
- Selecciona a pasta onde pretendes guardar a base de dados.
- Digita o nome da base de dados: **Alunos**.



2 Cria a tabela **Alunos** tendo em atenção as especificações referidas.

Especificações:

ALUNOS Número do aluno Nome Morada Código Postal Data de nascimento Sexo Telefone	NALUNO (Número do aluno) Tipo de dados: Sequencial (atribuição automática) Tamanho do campo: Inteiro longo Legenda: Número do aluno	NOME (Nome do aluno) Tipo de dados: Texto Tamanho do campo: 60 Obrigatório: Sim Permitir comprimento zero: Não Legenda: Nome do aluno
MORADA Tipo de dados: Texto Tamanho do campo: 50 Obrigatório: Sim Permitir comprimento zero: Não Indexado: Não	CPOSTAL (Código Postal) Tipo de dados: Texto Tamanho do campo: 4 Máscara de introdução: 9999 Legenda: Código Postal Obrigatório: Não Indexado: Não	DATANASC (Data de nascimento) Tipo de dados: Data/Hora Formato: Data abreviada Legenda: Data de nascimento Obrigatório: Sim Indexado: Não
SEXO Tipo de dados: Texto Tamanho do campo: 1 Regra de validação: M ou F Texto de validação: Introduza M para Masculino ou F para Feminino Obrigatório: Sim Permitir comprimento zero: Não Indexado: Não		TELEFONE Tipo de dados: Texto Tamanho do campo: 10 Máscara de introdução: (Escolha apropriada) Obrigatório: Não

Actividade n.º 1

- 1 Numa tabela, a que corresponde um registo? (ver pág. 9)
- 2 É fundamental, em cada tabela que constitui a base de dados, definir uma chave primária. Porquê? (ver pág. 12)
- 3 Quais as regras que uma chave primária deve respeitar?
- 4 Identifica e caracteriza os componentes essenciais de uma base de dados. (ver pág. 17)

Actividade n.º 2

- 1 Depois de analisares a figura, indica a função de cada uma das propriedades.
- 2 Indica a regra de validação adequada caso se pretenda que um campo definido como **Número** apenas possa assumir valores entre -10 e 10. (ver pág. 28)

Geral	Pesquisa
Tamanho do campo	
Formatar	
Máscara de introdução	
Legenda	
Valor predefinido	
Regra de validação	
Texto de validação	
Necessário	
Permitir comprimento zero	
Indexado	
Compressão Unicode	
Modo IME	
Modo de frase IME	
Etiquetas inteligentes	

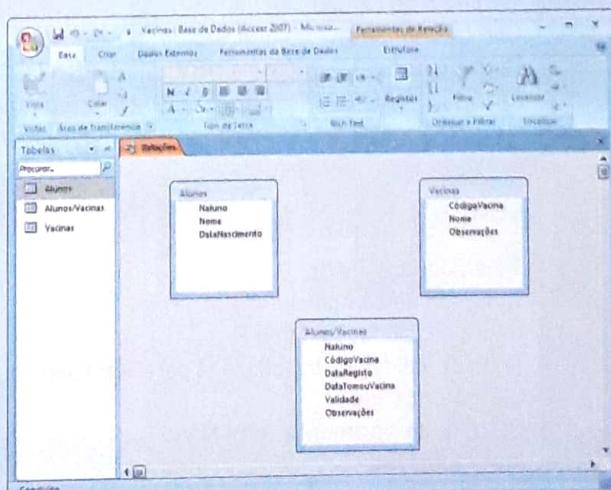
- 3 Indica a regra de validação adequada se pretendermos que um campo definido como **Data** não possa conter uma data superior à data actual (data do sistema).

Actividade n.º 3

Uma escola pretende ter o registo das vacinas dos seus alunos para verificar se as têm em dia e, caso tal não aconteça, os poder alertar para a necessidade de as tomar.

Neste sentido, considera as tabelas ilustradas na imagem.

- 1 Identifica as tabelas envolvidas e, para cada uma delas, indica o campo ou a combinação de campos que deveriam constituir a chave primária.
- 2 É fundamental relacionar os campos comuns das tabelas que constituem a base de dados. Porquê? (ver pág. 33)



Imprimir um relatório

1. Selecciona o relatório a imprimir.

2. No separador Base clica em Vista.

3. Selecciona a opção Pré-visualizar.

Assim, terás uma visão mais aproximada de como o relatório será impresso.

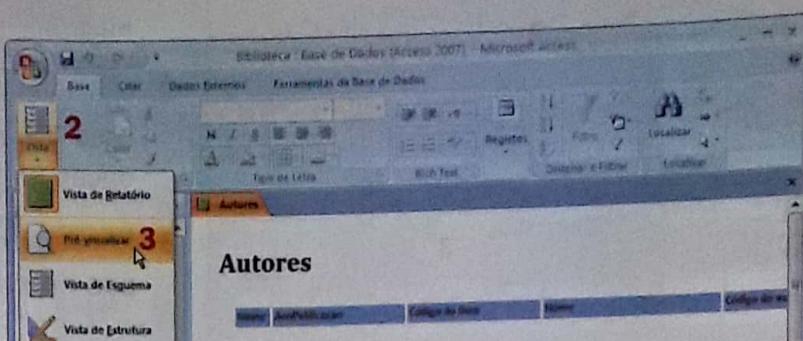


Fig. 105

4. No separador Pré-visualizar clica em Imprimir.

5. Selecciona as opções adequadas.

6. Clica em OK.

7. Clica em para fechar a Pré-visualização.

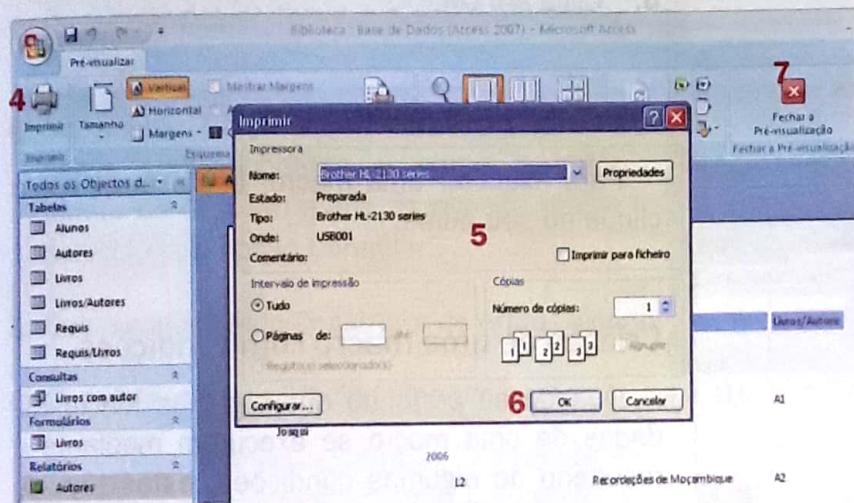


Fig. 106

3.14. Macros

Uma macro é um objecto do Access constituída por um conjunto de acções com o objectivo de executar tarefas específicas, nomeadamente abrir um formulário ou consulta, efectuar uma pesquisa, etc.

Depois de criada e gravada, uma macro pode ser executada sempre que necessário, permitindo, assim, a automatização de diversas tarefas.

3.13.2. Alterar um relatório

Mover campos

- Para movimentar os campos, é necessário remover o esquema predefinido. Para tal:
1. Selecciona os campos.
 2. Prime o botão direito do rato e, no menu de acesso rápido, selecciona a opção **Esquema**.
 3. Selecciona a opção **Remover**.
 4. Clica no campo a mover e arrasta-o para o local pretendido.

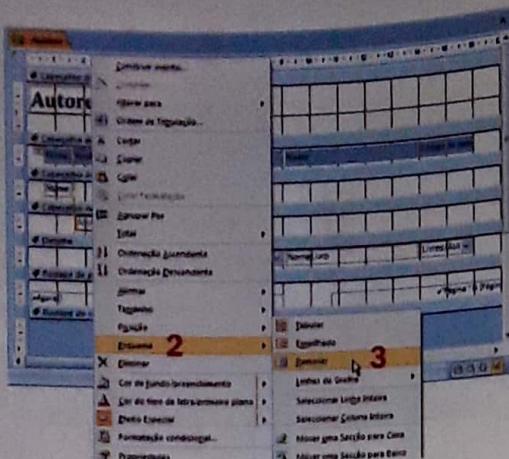


Fig. 102

Eliminar campos

Clica no campo a eliminar e prime a tecla **Del**.

Alinhar campos

Selecciona os campos a alinhar e, no menu de acesso rápido, selecciona a opção **Alinhar** seguida do alinhamento pretendido.

Outras formatações

Depois de seleccionares o(s) campo(s), analisa as opções de formatação disponíveis no separador **Estrutura**.

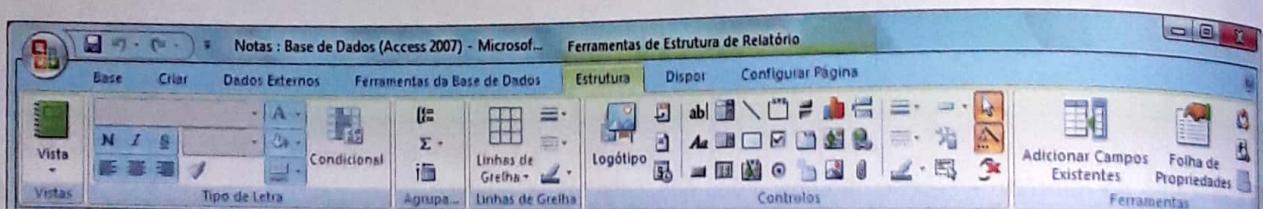


Fig. 103

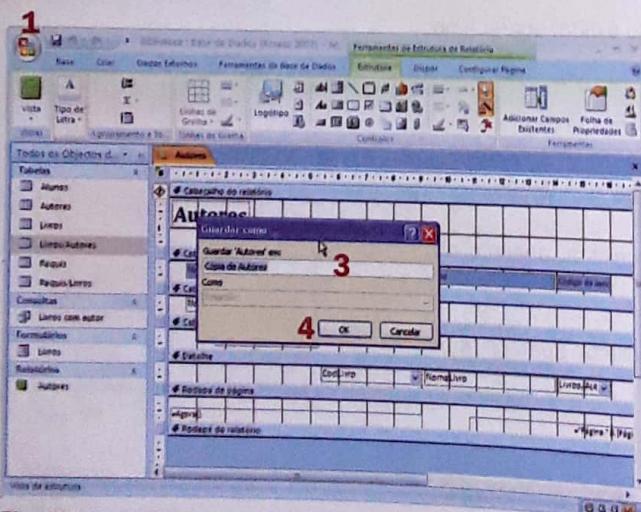


Fig. 104

Guardar um relatório

Para guardar as alterações efectuadas num relatório, basta clicar em .

Para guardar o relatório com outro nome:

1. Clica no **botão do Office**.
2. Selecciona a opção **Guardar como**.
3. Digita o nome que pretendes atribuir ao relatório.
4. Clica em **OK**.

Selecciona a tabela cuja informação pretendes visualizar. Neste caso, é a tabela **Livros**. Clica em **Seguinte**.

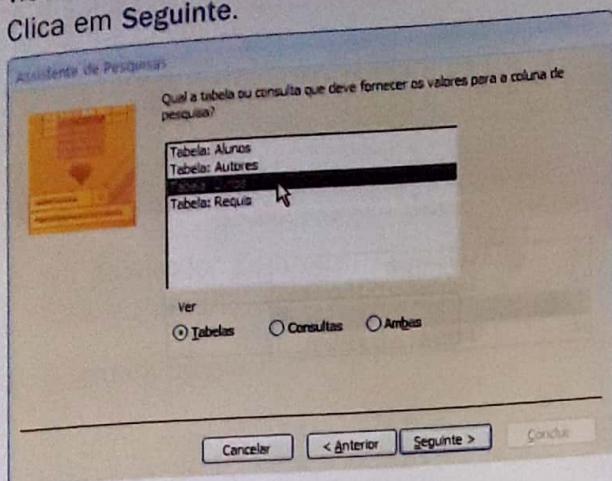


Fig. 40

Especifica a ordenação pretendida. Clica em **Seguinte**.

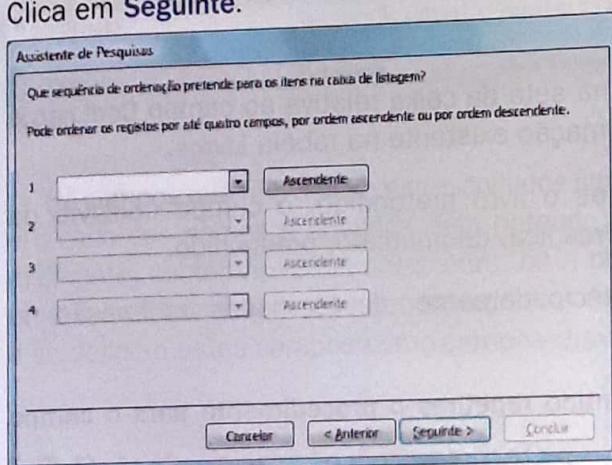


Fig. 42

Selecciona o campo cujo conteúdo pretendes guardar. Clica em **Seguinte**.

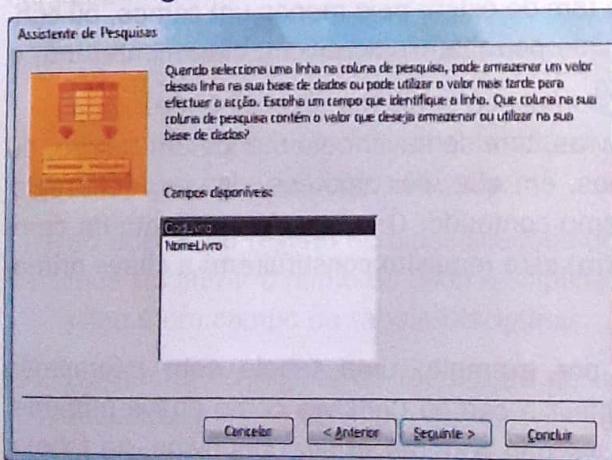


Fig. 44

Selecciona os campos a visualizar e, fazendo duplo clique ou utilizando os botões, move-os para a caixa **Campos seleccionados**. Clica em **Seguinte**.

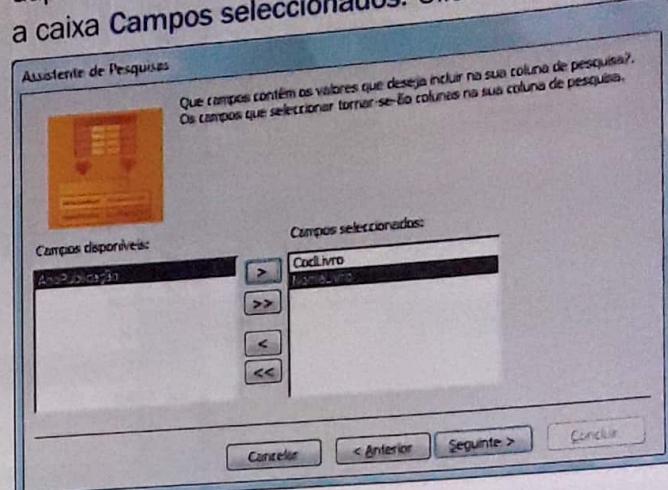


Fig. 41

Define a largura dos campos visualizados. Clica em **Seguinte**.

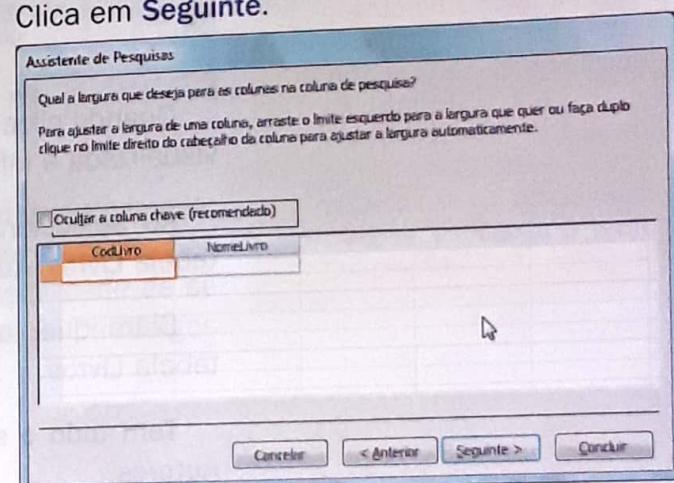


Fig. 43

Como etiqueta é sugerido o nome do campo, mas podes digitar outro texto. Clica em **Concluir**.

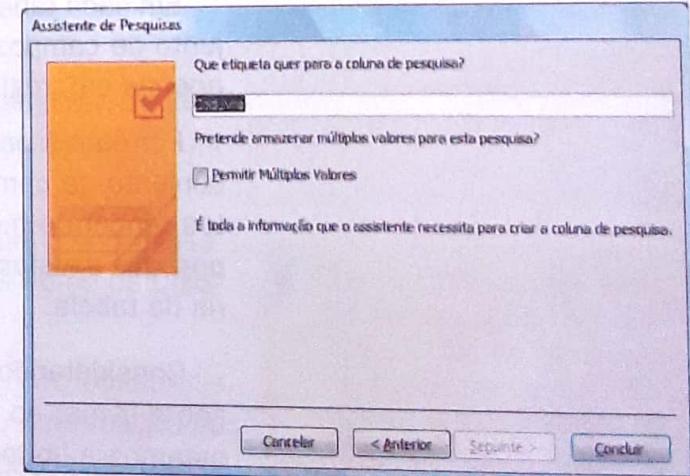


Fig. 45

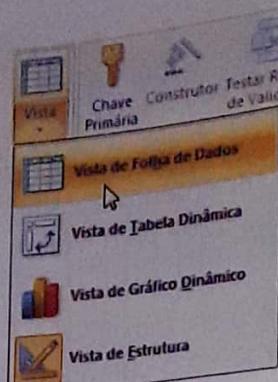


Fig. 36

- Para verificares o efeito desta funcionalidade:
- clica em para que as alterações efectuadas à tabela sejam gravadas;
 - no botão **Vistas**, clica em **Vista de Folha de Dados** para poderes introduzir dados na tabela (Fig. 36).

Analisando a figura, no campo **ClasseAluno** é dada a possibilidade de optar por 10, 11 ou 12 (Fig. 37).

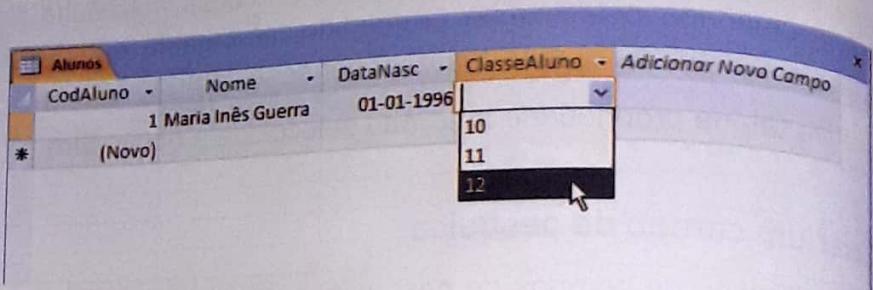


Fig. 37

Podes, também, criar um campo de pesquisa em que os valores visualizados na lista pertencem a uma tabela.

Por exemplo, para introduzir os códigos dos livros e os autores na tabela **Livros/Autores**, seria útil aceder às respectivas tabelas.

No caso dos **Livros**, vamos analisar o procedimento a seguir.

No campo **CodLivro**, em **Tipo de dados**, seleciona a opção **Assistente de Pesquisas**.

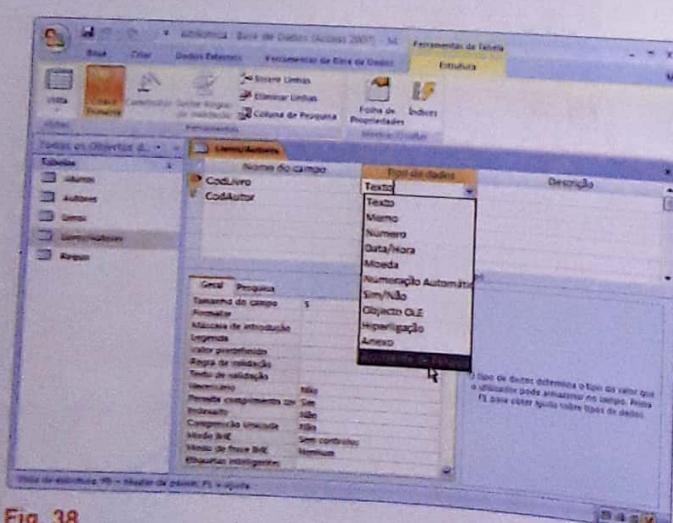


Fig. 38

Seleciona a opção **Quero que a coluna de pesquisa pesquise os valores numa tabela ou consulta**. Clica em **Seguinte**.

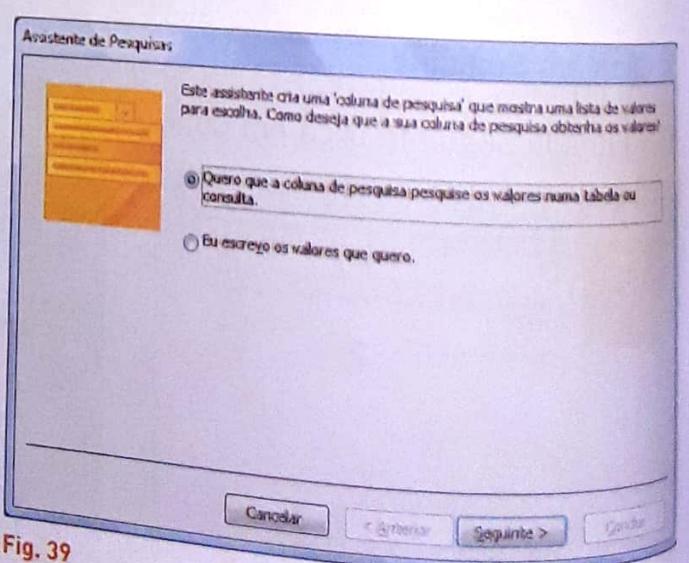


Fig. 39

Permitir comprimento zero

Esta propriedade só se aplica a campos do tipo Texto, Memo e Hipérlinkação.

Uma cadeia de comprimento zero pode ser entendida como uma cadeia que não contém caracteres e é utilizada para indicar que um campo não tem qualquer valor.

Para introduzir uma cadeia de comprimento zero, digita duas aspas ("") seguidas. Podes assim determinar se o campo não foi preenchido ou não tem conteúdo.

Indexado

A criação de um índice faz com que o Access mantenha a tabela ordenada pelo campo pretendido, permitindo optimizar todas as pesquisas que se realizem por esse campo.

Para tal, na propriedade **Indexado** selecciona a opção Sim.

Criar um campo de pesquisa

A utilização de campos de pesquisa, sempre que adequada, facilita a tarefa de introdução de dados num campo, pois permite seleccionar valores existentes numa lista.

Para tal:

1. Define o **Tipo de dados** do campo como **Assistente de Pesquisas**.
2. Na janela **Assistente de Pesquisas** activa a opção **Eu escrevo os valores que quero.**
3. Clica em **Seguinte.**
4. Digita a lista de valores a visualizar.
5. Clica em **Seguinte.**
6. Na janela seguinte clica em **Concluir.**

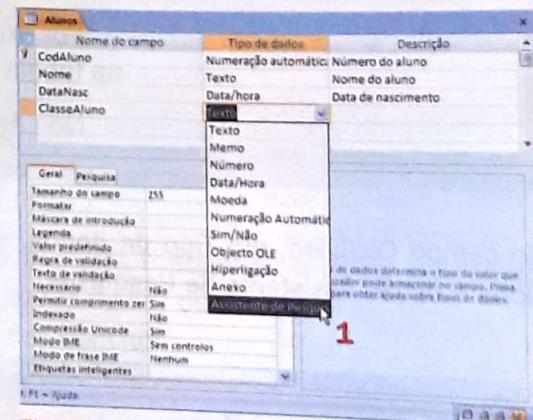


Fig. 33

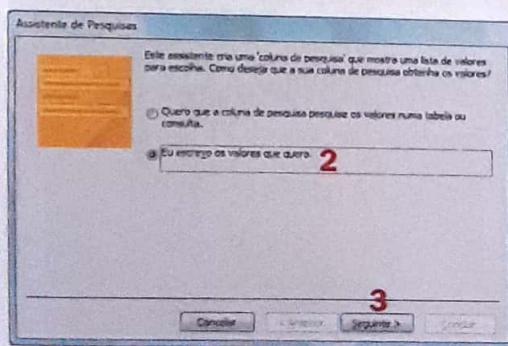


Fig. 34

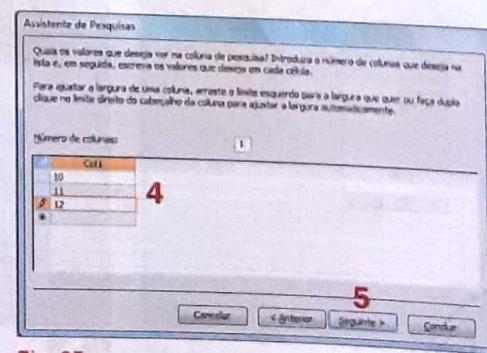


Fig. 35

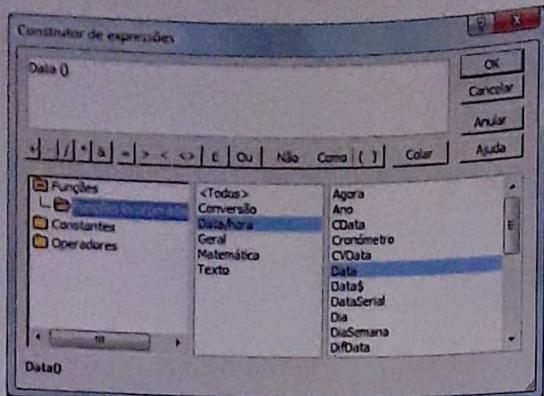


Fig. 29

Contudo, para além de constantes, também podes utilizar expressões. Para tal, clica em para aceder ao **Construtor de expressões** e definires a expressão adequada.

Supõe que pretendes que um campo do tipo **Data** tenha, como valor predefinido, a data do sistema. Para tal, deves aceder ao **Construtor de expressões** e seleccionar as opções, tal como na figura 29.

Geral	Pesquisa
Tamanho do campo	Número inteiro longo
Formatar	
Casas decimais	Automático
Máscara de introdução	
Legenda	
Valor predefinido	
Regra de validação	<code><=2013</code>
Texto de validação	
Necessário	Não
Indexado	Não
Etiquetas inteligentes	
Alinhamento do texto	Geral

Fig. 30

Regra de validação

Para evitar eventuais erros de digitação, é conveniente que, sempre que adequado, seja feita a validação do campo, especificando a regra a que o seu conteúdo deve obedecer.

Para tal, na propriedade **Regra de validação** digita a regra pretendida ou clica no botão para aceder ao **Construtor de expressões** e criar a regra de validação adequada.

Supõe que pretendes validar um campo numérico para que não sejam aceites valores superiores a 2013.

No campo **Regra de validação** basta digitar `<=2013`.

Geral	Pesquisa
Tamanho do campo	Número inteiro longo
Formatar	
Casas decimais	Automático
Máscara de introdução	
Legenda	
Valor predefinido	
Regra de validação	<code>Entre 10 E 25</code>
Texto de validação	
Necessário	Não
Indexado	Não
Etiquetas inteligentes	
Alinhamento do texto	Geral

Fig. 31

Supõe que pretendes validar um campo numérico para que apenas sejam aceites valores entre 10 e 25, inclusive.

No campo **Regra de validação** basta digitar `Entre 10 E 25`.

Geral	Pesquisa
Tamanho do campo	40
Formatar	
Máscara de introdução	
Legenda	Nome do aluno
Valor predefinido	
Regra de validação	
Texto de validação	
Necessário	Sim
Permitir comprimento zero	Sim
Indexado	Sim
Compressão Unicode	Sim
Modo IME	Sem controlos
Modo de frase IME	Nenhum

Fig. 32

Necessário

Há campos que, pela sua importância, devem ser preenchidos. Para garantir que tal acontece, no campo **Necessário** deves seleccionar a opção **Sim**.

Por exemplo, na tabela Alunos não deveriam existir registo sem o nome do aluno. Logo, o campo relativo ao nome do aluno deve ser definido como de **preenchimento obrigatório**.

Propriedade	Função	Tipo de dados a que se aplica
Tamanho do campo	Número máximo de caracteres ou números que o campo pode conter.	Numeração automática, texto, número e data/hora
Novos valores	Determinar se um campo é incrementado em cada novo registo ou se é gerado um número aleatório.	Numeração automática
Formatar	Formato em que textos, números ou datas serão visualizados.	Numeração automática, texto, memo, número, moeda, sim/não e hiperligação
Casas decimais	Número de casas decimais que devem ser visualizadas.	Número e moeda
Máscara de introdução	Especificação de uma máscara, ou molde, para controlo de entrada dos dados.	Texto, número e moeda
Legenda	Etiqueta do campo que o identifica em modo de Vista de Folha de Dados e é apresentada em formulários, relatórios e consultas. Se esta propriedade estiver vazia, é utilizado o próprio nome do campo.	Numeração automática, texto, memo, número, moeda, data/hora, sim/não, objecto OLE e hiperligação
Valor predefinido	Valor que o campo assumirá, por defeito, aquando da criação de um novo registo mas que pode ser alterado.	Texto, memo, número, moeda, data/hora, sim/não e hiperligação
Regra de validação	Critério ou condição para aceitação dos dados no campo.	Texto, memo, número, moeda, data/hora, sim/não e hiperligação
Texto de validação	Texto visualizado no ecrã quando a regra de validação não é respeitada.	Texto, memo, número, moeda, data/hora, sim/não e hiperligação
Necessário	Se "Sim" implica que o campo seja de digitação obrigatória (tem de ter conteúdo).	Texto, memo, número, moeda, data/hora, objecto OLE e hiperligação
Permitir comprimento zero	Permitir (ou não) que o campo possa ser vazio.	Texto, memo e hiperligação
Indexado	Indicar se o campo está, ou não, indexado. Quando um campo está indexado, as pesquisas ao seu conteúdo são mais rápidas.	Numeração automática, texto, memo, número, moeda, data/hora, sim/não e hiperligação
Compressão Unicode	Indica se o conteúdo do campo pode, ou não, ser comprimido através do <i>standard Unicode</i> (útil quando é armazenada grande quantidade de texto).	Texto, memo e hiperligação
Modo IME	Controlar a conversão de caracteres nas versões asiáticas do Windows.	Texto, memo, data/hora e hiperligação
Modo de frase IME	Controlar a conversão de frases nas versões asiáticas do Windows.	Texto, memo, data/hora e hiperligação
Etiquetas inteligentes	Anexar uma etiqueta inteligente ao campo (dados reconhecidos como pertencendo a um determinado tipo).	Numeração automática, texto, memo, número, moeda, data/hora e hiperligação
Mostrar seleccionador de datas	Especificar se o controlo seleccionador de datas é ou não apresentado.	Data/hora
Alinhamento do texto	Especificar o alinhamento predefinido do texto num controlo.	Numeração automática, memo, número, moeda, data/hora, sim/não, objecto OLE e hiperligação

Tamanho do campo

Permite definir o número máximo de caracteres que o campo poderá aceitar.

Se o campo for do tipo texto, pode conter entre 0 e 255 caracteres. Tratando-se de um campo numérico, depende da opção escolhida.

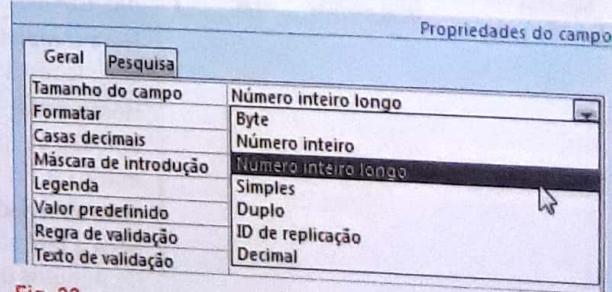


Fig. 23

Tipo de dados que podemos especificar

Tipo de dados	Descrição (o que o campo poderá conter)
Texto	Caracteres alfanuméricos (máx. de 255 caracteres).
Memo	Caracteres alfanuméricos (normalmente utilizado em campos relativos a comentários ou observações).
Número	Valores numéricos (inteiros ou fracionários).
Data/Hora	Datas e horas válidas.
Moeda	Valores monetários.
Numeração Automática	Valor numérico definido automaticamente pelo Access.
Sim/Não	Valores booleanos (quando o campo apenas pode ter uma de duas opções). Ex.: Sim/Não; Verdadeiro/Falso.
Objecto OLE	Documentos do Microsoft Word, folhas de cálculo do Microsoft Excel, imagens, sons ou outros dados binários criados noutros programas que utilizem o protocolo OLE, que podem ser ligados ou incorporados numa tabela do Microsoft Access.
Hiperligação	Ligaçāo que permite abrir um ficheiro ou página Web.
Anexo	Indicação da localização e nome de um ou mais ficheiros que, posteriormente, poderão ser editados.
Assistente de Pesquisa	Definição de um campo que permite seleccionar um valor existente numa tabela ou numa caixa de combinação.

Propriedades dos campos

Todos os campos que constituem uma tabela têm propriedades inerentes ao tipo de campo.

As propriedades permitem, por exemplo, definir o formato de introdução de um valor no campo, se o seu preenchimento é, ou não, obrigatório, etc.

Exemplos:

Propriedades que podem ser definidas para um campo do tipo **Texto**.

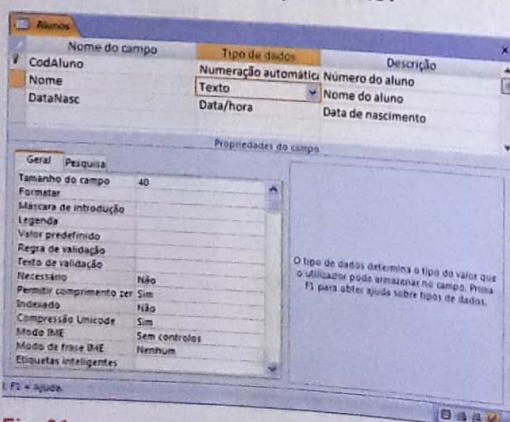


Fig. 21

Propriedades que podem ser definidas para um campo do tipo **Data/Hora**.

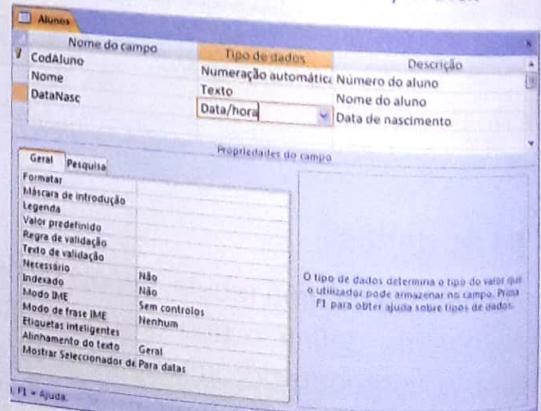


Fig. 22

Para tal:

1. Acede ao Access.
2. Na janela Microsoft Access acede à base de dados onde pretendes criar a tabela.
3. No separador Criar, grupo Tabela, clica em Tabela.
4. Clica em Vista e, em seguida, em Vista de Estrutura.
5. Na janela Guardar como digita o nome que pretendes atribuir à tabela.
6. Clica em OK.
7. Em Nome do campo digita o nome que pretendes atribuir ao campo.
8. Em Tipo de dados selecciona o tipo adequado ao campo.
9. Em Descrição digita um texto que, de alguma forma, descreva o campo.
10. Especifica o Tamanho do campo.

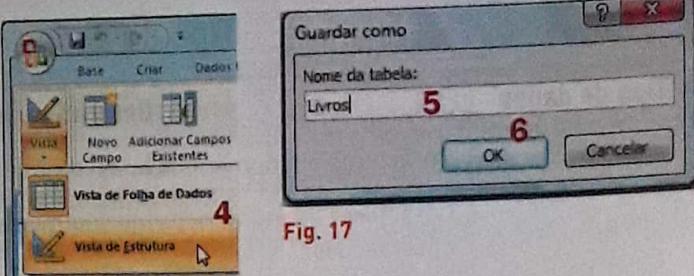


Fig. 17

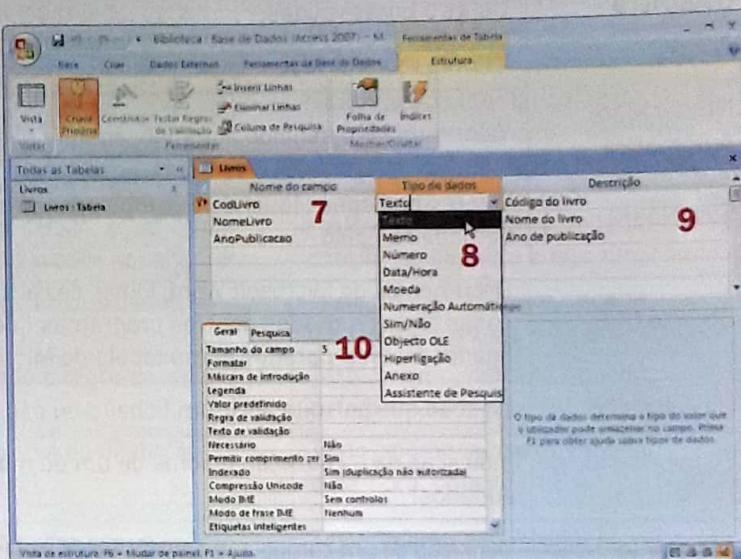


Fig. 18

Repete os pontos 7, 8, 9 e 10 para os restantes campos que constituam a tabela.

11. Clica em para guardar a tabela.

É visualizada uma janela com indicação de que nenhuma chave primária foi definida. Como, neste momento, ainda não tens informação suficiente para efectuares esta operação, clica em Não.

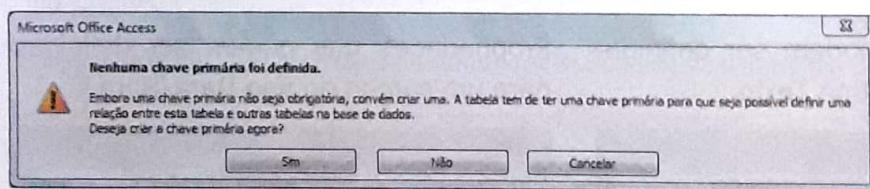


Fig. 19

Tipo de dados

Na criação de uma tabela, depois de definir o nome do campo, é necessário definir o tipo de dados que o campo irá conter. Assim, o conteúdo de um campo é condicionado pelo **Tipo de dados** definido. Quando um campo é criado, assume, automaticamente, o tipo de dados **Texto**.

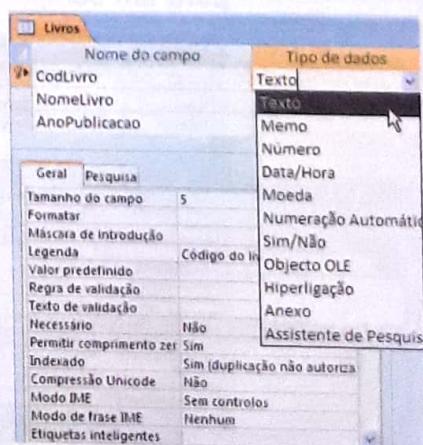


Fig. 20

3.5. Tabelas

Criar tabelas com base num modelo

O Access 2007 disponibiliza alguns modelos pelos quais podes optar para criar, automaticamente, uma tabela.

Para tal:

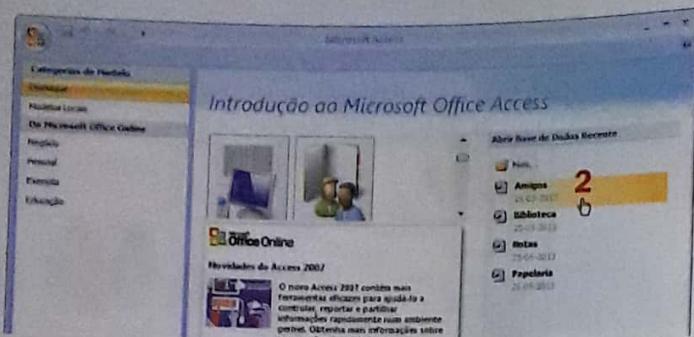


Fig. 13

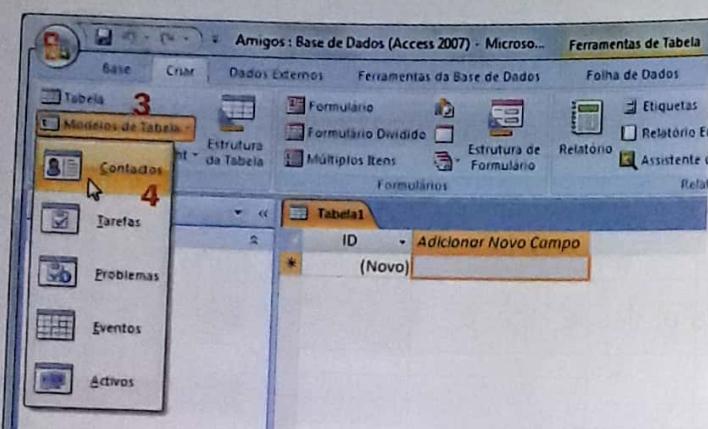


Fig. 14

1. Acede ao Access.

2. Na janela Microsoft Access acede à base de dados onde pretendes criar a tabela.

Se o nome da base de dados estiver visível, basta clicar sobre o seu nome. Senão, clica em Mais... para acederes à janela Abrir e, depois de seleccionada a drive e a pasta, selecciona a base de dados pretendida.

3. No separador Criar, grupo Tabela, clica em Modelos de Tabela.

4. Selecciona o modelo pretendido.

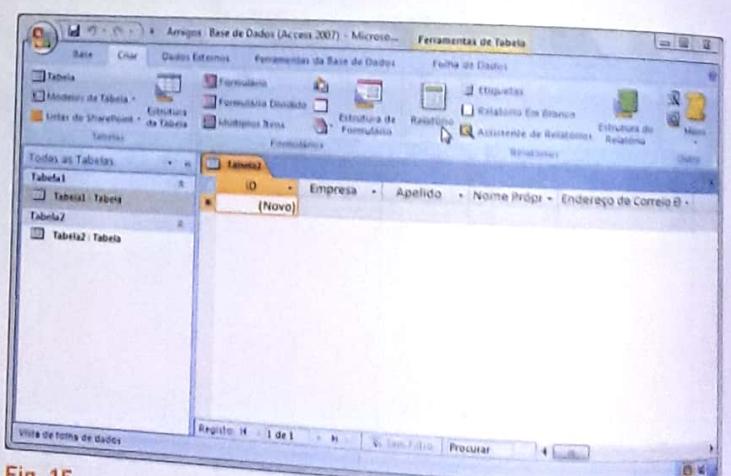


Fig. 15



A tabela é gravada com os campos visualizados e, desde logo, podes inserir informação.

Criar tabelas em Vista de Estrutura

Este é o método mais utilizado para criar tabelas, pois permite-nos definir apenas os campos que necessitamos, bem como a sua estrutura.

3.3. Elementos de uma base de dados

Tabelas

Uma tabela é constituída por um conjunto de dados sobre um determinado tema ou assunto, organizados em colunas (campos) e linhas (registos).

Exemplos de tabelas: Filmes, Actores, Alunos, etc.

Consultas

As consultas permitem visualizar os dados existentes em tabelas e/ou outras consultas, normalmente segundo condições pré-estabelecidas. A informação gerada numa consulta também pode ser utilizada como origem de registos para formulários e relatórios.

Formulários

Fundamentalmente, os formulários são utilizados como interface para facilitar a introdução de dados.

Relatórios

Os relatórios permitem apresentar os dados em formato impresso.

Macros

As macros permitem automatizar a realização das tarefas que executamos com maior frequência. Esta funcionalidade também existe no Word e no Excel.

Módulos

São declarações, instruções e procedimentos do **Visual Basic**, guardados em conjunto, que permitem efectuar alguns procedimentos específicos (validações, cálculos, etc.).



O Visual Basic é uma linguagem de programação desenvolvida pela Microsoft.

3.1. A janela do programa

Quando se acede ao Office Access 2007 é apresentada a página de Introdução ao Microsoft Office Access a partir da qual podes começar a trabalhar.

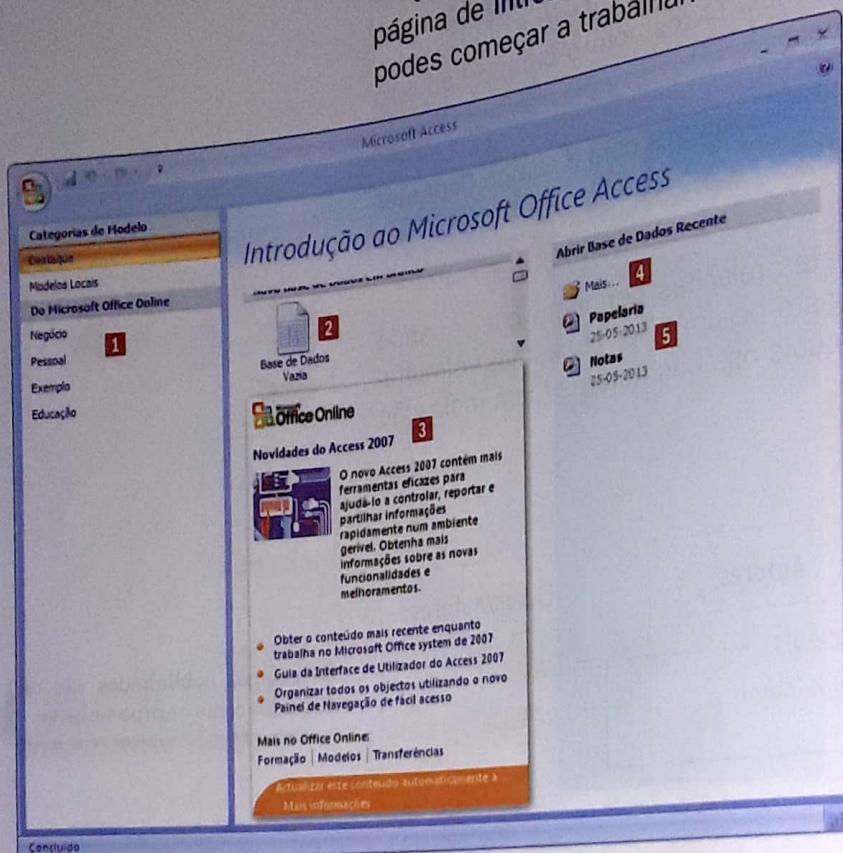


Fig. 7

3.2. O ambiente de trabalho

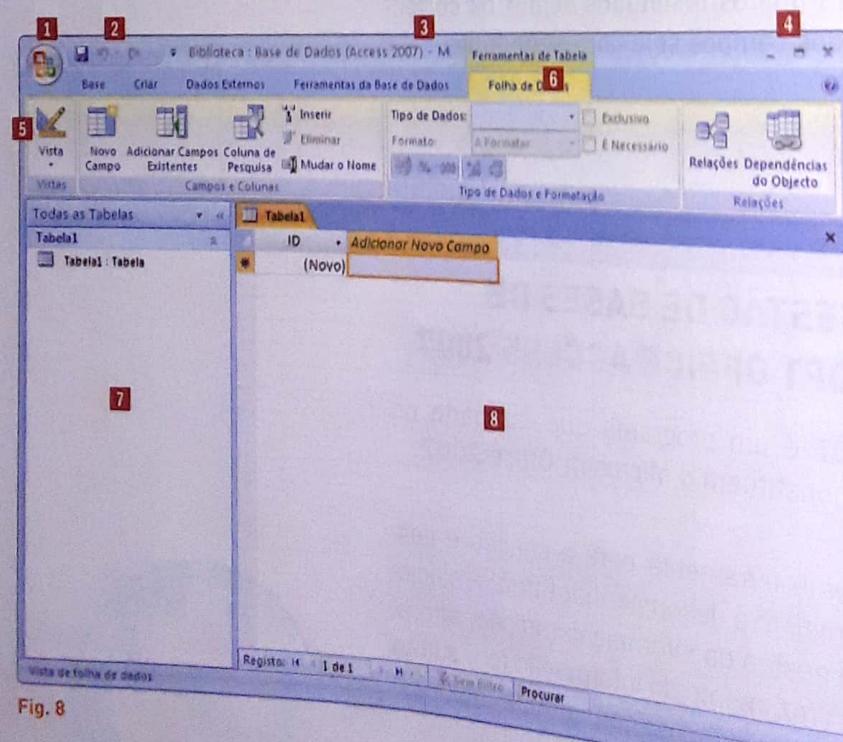


Fig. 8

- 1 Criar uma base de dados a partir de um modelo existente.
- 2 Criar uma nova base de dados.
- 3 Consultar o Microsoft Office Online.
- 4 Aceder a uma base de dados existente.
- 5 Aceder a uma base de dados aberta recentemente.

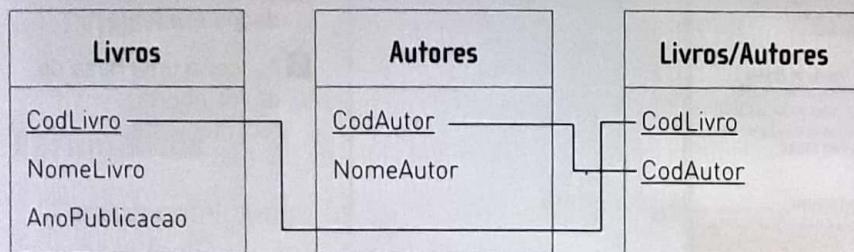
- 1 Botão do Office.
- 2 Barra de ferramentas de acesso rápido.
- 3 Barra de título.
- 4 Botões minimizar, maximizar e fechar.
- 5 Friso constituído por um conjunto de separadores com comandos, organizados por grupos.
- 6 Separadores do friso.
- 7 Painel de navegação.
- 8 Área de trabalho.

Definir as relações entre as tabelas:

As relações devem ser estabelecidas tendo em atenção os campos comuns das tabelas. Assim, podemos relacionar:

- o campo **CodLivro** da tabela **Livros** com o campo **CodLivro** da tabela **Livros/Autores**;
- o campo **CodAutor** da tabela **Autores** com o campo **CodAutor** da tabela **Livros/Autores**.

Depois de definidas, estas relações permitem que, quando estamos a visualizar o registo de um livro, possamos, simultaneamente, saber qual ou quais os seus autores (com indicação dos respectivos nomes).



Os campos sublinhados são os definidos como **campos-chave**.

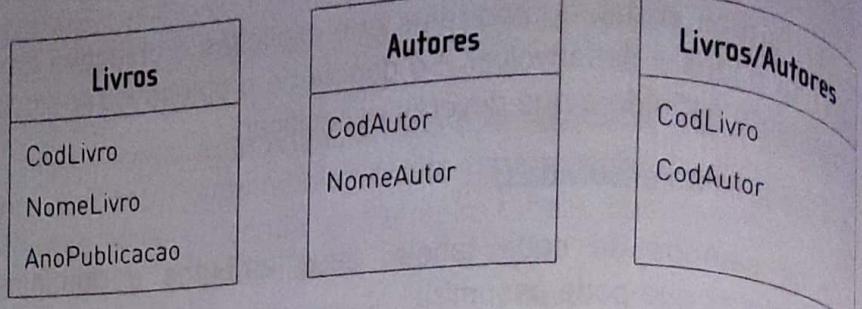
Neste exemplo, os campos relacionados têm o mesmo nome, mas não é obrigatório. Para que os resultados sejam os corretos, basta que o conteúdo dos campos seja correspondente.

3. PROGRAMA DE GESTÃO DE BASES DE DADOS: MICROSOFT OFFICE ACCESS 2007

O Microsoft Access 2007 é um programa que faz parte do pacote de aplicações que constituem o Microsoft Office 2007.

Considerada uma excelente ferramenta para a criação e gestão de bases de dados, permitir-te-á desempenhar tarefas relacionadas com a manutenção e gestão da informação (adição, alteração, consulta e eliminação), cruzamento de informação (consultas personalizadas) e elaboração de relatórios, com grande rapidez e eficácia.

Definir o nome das tabelas e respectivos campos:



Definir os atributos de cada campo bem como o seu domínio:

Campo	Descrição	Tipo	Tamanho	Domínio
CodLivro	Código do livro	Texto	13	Sem restrições
NomeLivro	Nome do livro	Texto	50	Sem restrições
AnoPublicacao	Ano de publicação do livro	Número inteiro longo	4	Inferior ou igual ao ano actual
CodAutor	Código do autor	Texto	3	Sem restrições
NomeAutor	Nome do autor	Texto	50	Sem restrições

Definir os campos-chave:

- na tabela **Livros** deve ser definido o campo **CodLivro**, pois este permite identificar apenas um livro. Não pode haver mais do que um livro com o mesmo código;
- na tabela **Autores** deve ser definido o campo **CodAutor**, pois este campo permite identificar apenas um autor. Não pode haver mais do que um autor com o mesmo código;
- na tabela **Livros/Autores** devem ser definidos os campos **CodLivro** e **CodAutor**. Isto porque:
 - se o livro tiver mais do que um autor, o seu código irá repetir-se. Não pode, portanto, sozinho, ser chave;
 - se o mesmo autor tiver mais do que um livro editado, o seu código irá repetir-se. Não pode, portanto, sozinho, ser chave.

Efectivamente, a combinação **CodLivro-CodAutor** permite identificar um e apenas um registo.

2. ORGANIZAÇÃO DE UMA BASE DE DADOS

Numa fase inicial, é importante que explicites o objectivo da base de dados a desenvolver. Só depois de o objectivo ter sido claramente definido é que deverás especificar:

- as tabelas envolvidas;
- os campos de cada tabela, seus atributos e domínio (valores que pode assumir);
- o(s) campo(s)-chave;
- o relacionamento entre os campos das tabelas;
- as características das relações.

Analisa a seguinte situação prática:

Imagina que tens muitos livros e gostarias de criar uma base de dados para que, de forma simples e rápida, possas saber os livros que tens, o seu ano de edição e os seus autores.

Objectivo da base de dados: Saber os livros que tens e quais os seus autores.

Num registo manual, em forma de tabela, terias, por exemplo:

Nome do livro	Ano de publicação	Autor
Mar Me Quer	2001	Mia Couto
Recordações de Moçambique	2006	Joaquim Carlos Vieira Ana Paula Lemos
Gotas de Sol	2006	Calane da Silva
Balada do amor ao vento	1990	Paulina Chiziane

Pressupostos:

- podes ter vários livros do mesmo autor – para não digitares mais do que uma vez o nome de um autor, deves criar uma tabela de autores;
- cada livro pode ter mais do que um autor – como o número de autores dos livros é variável, deves criar uma tabela para o seu registo (autores de cada livro);
- os livros e os autores devem ser identificados por um código.

Nesta situação, um livro pode ser requisitado várias vezes e, portanto, o seu código constar em várias requisições. Por outro lado, cada requisição pode ser de diversos livros e, portanto, ser constituída por vários códigos de livros.

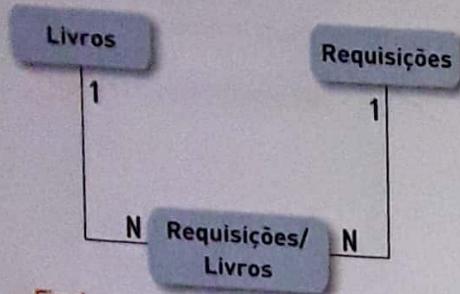
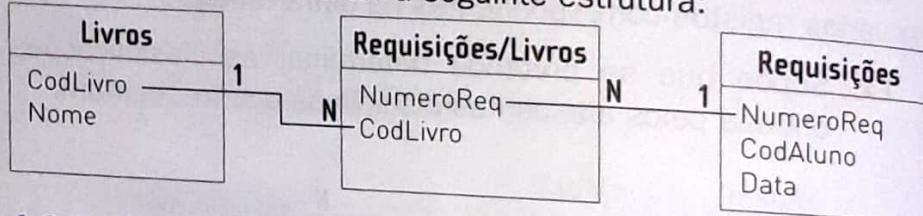


Fig. 6

Em Access, esta situação não é desejável e, portanto, sempre que entre duas tabelas exista uma relação de muitos-para-muitos, deve ser criada uma terceira tabela, chamada tabela de associação, e estabelecida uma relação entre as tabelas iniciais e a **tabela de associação**. Assim, deixa de existir uma relação de muitos-para-muitos e passamos a ter duas relações de um-para-muitos.

As tabelas poderiam ter a seguinte estrutura:



1.3.5. Noção de chave

A **chave** de uma tabela é constituída por um ou mais campos que possam ser utilizados como identificadores de cada um dos registo.

Na seleção do(s) campo(s) deves ter em atenção o seguinte:

- o campo, ou campos, que constituem a chave devem permitir identificar um registo de forma unívoca (um e apenas um registo);
- o campo, ou campos, que constituem a chave não podem conter um valor nulo em nenhum registo (são sempre de digitação obrigatória);
- no caso de ser necessário que a chave seja constituída por mais do que um campo (chave composta), não devem ser incluídos mais campos do que os estritamente necessários para identificar os registo.

Ex.:

Tabela	Campo(s)-chave
Alunos	Código do aluno
Classificações	Código do aluno + código da disciplina
Livros	Código do livro
Requisições	Número da requisição
Requisições/Livros	Número da requisição + código do livro

Numa tabela onde se pretende guardar a informação dos alunos de uma escola, o campo relativo à **Data de nascimento** deveria ser escolhido para chave?

A resposta é **não**. Com certeza que haverá vários alunos que nasceram no mesmo dia e, sendo assim, o campo **Data de nascimento** teria valores repetidos. Mesmo que, de momento, não existissem alunos com a mesma data de nascimento, esta possibilidade teria sempre de ser considerada.

O relacionamento de campos comuns de duas tabelas permite aceder, simultaneamente, a todos os outros campos que constituem os registo.

1.3.3. Noção de relação

Uma relação consiste na associação estabelecida entre campos comuns de duas entidades, garantindo a consistência da informação.

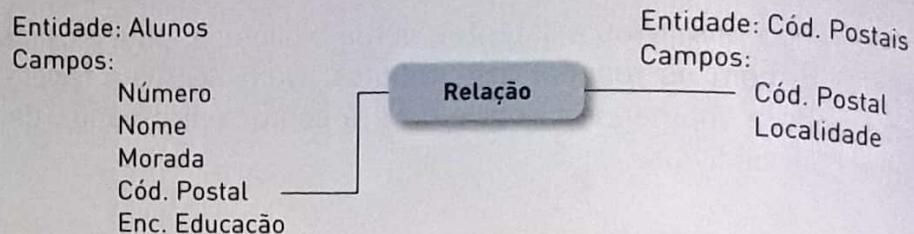


Fig. 3

Transportando para um SGBD, temos:

Tabela: Alunos

Número	Nome	Morada	Cód. Postal	Enc. Educação
1	José Ferreira	Rua Direita	1100	Paulina Ferreira
2	Noémia Santos	Rua Central	1101	Sérgio Santos
3	Reinaldo Guerra	Av. 24 de Julho	1100	Maria Guerra

Tabela: Cód. Postais

Cód. Postal	Localidade
1100	Maputo1
1101	Polana

A relação estabelecida entre o campo **Cód. Postal** da tabela **Alunos** e o campo **Cód. Postal** da tabela **Cód. Postais** permitirá que:

- na tabela **Alunos**, o campo **Localidade** deixe de existir e, portanto, não tenha de ser digitado em todos os registo;
- na tabela **Alunos**, seja validado que o Cód. Postal existe na tabela **Cód. Postais**;
- na consulta ou impressão dos dados dos alunos, seja possível visualizar a localidade correspondente ao código postal;
- seja minimizado o número de erros, pois na tabela **Alunos** apenas é digitado o Cód. Postal e, regra geral, cometem-se menos erros na digitação de números do que de texto.



A diferença num carácter, seja um espaço, uma vírgula, um ponto final ou qualquer outro, faz com que o conteúdo de dois campos seja considerado diferente. Por exemplo, **Matola-Rio** e **Matola Rio** seriam consideradas localidades diferentes.

Desde os tempos remotos da sua existência, o homem lutou sempre para criar facilidades na sua vida e o levou a inventar engenhos simples ou complexos que o auxiliem nos seus trabalhos, com o objectivo de reduzir o tempo e esforço. Das várias máquinas que o homem criou, actualmente destaca-se o computador, ferramenta indispensável nas sociedades modernas, pois é bastante versátil devido à sua abrangência, segurança e velocidade.

Assim, a aquisição de conhecimentos e o desenvolvimento de competências nesta matéria é fundamental!

Neste sentido, a disciplina de TIC da 12.º classe tem como objectivo preparar os alunos para que sejam capazes de:

- dominar os conceitos básicos de bases de dados, programação e segurança;
- criar e usar bases de dados no tratamento e armazenamento da informação;
- conhecer as principais fases da elaboração de um projecto e saber implementá-lo;
- aplicar os conhecimentos sobre a segurança de computadores;
- avaliar convenientemente um projecto.

Estes conhecimentos, complementares aos adquiridos na 11.º classe, tornarão o aluno apto a:

- desenvolver a capacidade de aprender ao longo da vida para resolver problemas do quotidiano, usando as TIC;
- fomentar o interesse pela pesquisa, pela descoberta e pela inovação;
- desenvolver a capacidade de pesquisar, tratar, produzir e comunicar informação, quer por tecnologias tradicionais, quer através das novas tecnologias de informação e comunicação;
- desenvolver capacidades para utilizar adequadamente e manipular com rigor técnico aplicações informáticas articulando com as aprendizagens e tecnologias das outras áreas de conhecimento;
- promover as práticas inerentes às normas de segurança dos dados e da informação;
- utilizar as TIC no processo de interacção com a comunidade para o levantamento de necessidades, ouvir opiniões e propor projectos;
- lidar com a complexidade, diversidade e mudança;
- desenvolver projectos e estratégias de implementação individualmente ou em grupo.

Este livro pretende ser um suporte simples mas consistente para as tuas aprendizagens e desenvolvimento de competências, na área das Tecnologias de Informação e Comunicação.

Desejamos que te ajude!

À data da publicação deste livro, os sites referidos encontravam-se activos e os seus conteúdos tratavam assuntos desenvolvidos no âmbito das diferentes temáticas anunciadas.

Dada a grande dinâmica e elevada rapidez de evolução inerente à Internet, a Plural Editores não pode ser responsabilizada por qualquer desactivação ou alteração de conteúdo. Contudo, a Plural Editores assume toda a responsabilidade relativamente aos conteúdos constantes nos seus próprios sites.