

Il Pascal

Università degli Studi di Salerno
Corso di Laurea in Scienze della Comunicazione
Informatica generale (matr. Dispari)
Docente: [Angela Peduto](#)
A.A. 2005/2006



- Un consiglio: per integrare e quindi diventare padroni della materia c'è bisogno di esercitarsi sul computer e scatenare la fantasia nel creare programmi all'apparenza inutili, anche per le cose più banali, a lungo andare risulteranno utili per entrare nella logica di programmazione.
- Una volta educato il cervello a ragionare in una certa direzione, infatti, sarà facile imparare anche altri linguaggi di programmazione.

Programmazione



- Il computer è una macchina che elabora delle informazioni;
- in quanto macchina, però, non è capace di gestire le informazioni da se, ma necessita di un operatore esterno che 'istruisca l'elaboratore e gli insegni come trattare queste informazioni'.
- Questa definizione racchiude in se il significato della parola programmazione.

Interpreti e compilatori



- Esistono due principali tipi di linguaggi con cui programmare una macchina:
 - interpreti
 - compilatori

Interpreti e compilatori (2)



- Alla prima categoria appartengono numerosi programmi tra i quali spicca il BASIC (Beginners All-Purpose symbolic Instruction Code = Codice simbolico di istruzioni per tutti gli usi, per principianti) grazie alla sua intuitività.
- Il suo funzionamento è semplice: il codice sorgente (quello introdotto dall' operatore esterno) viene tradotto in linguaggio macchina (ASCII) riga per riga ogniqualvolta il programma stesso viene eseguito.

Interpreti e compilatori (3)



- I programmi compilatori funzionano nel modo opposto degli interpreti:
- dato il solito codice sorgente, il compilatore lo trasforma in blocco in codice eseguibile.

Interpreti e compilatori (4)



- Questi due metodi di programmazione sono entrambi validi ma ciascuno offre vantaggi e svantaggi.
- I compilatori infatti hanno il vantaggio di essere più veloci rispetto agli interpreti ma sono più soggetti ad errori degli interpreti

Introduzione al Pascal



- In questa sede ci occuperemo estesamente di un particolare tipo di linguaggio di programmazione compilato, il **Pascal**.
- Il Pascal prende il suo nome dal matematico francese Blaise Pascal che fu il primo ad ideare una macchina calcolatrice: la Pascalina.
- Questo linguaggio, però fu messo a punto nel 1960 da Niklaus Wirth, docente all' università di Zurigo.

Struttura del Programma



- Un programma scritto in Pascal consta di tre sezioni principali:
 - **sezione intestazione**
 - **sezione dichiarazioni**
 - **sezione esecutiva**
- nella prima sezione si inserisce il nome del programma,
- nella seconda sezione vengono digitate le variabili e la loro tipologia
- infine viene scritto il programma vero e proprio

Struttura del Programma (2)



- Per esempio possiamo considerare il Pascal come un teatro in cui si inscena un'opera:
 - il titolo dell' opera è dato dalla sezione delle intestazioni,
 - gli attori che prendono parte all' opera vengono presentati nella sezione dichiarativa,
 - mentre lo svolgimento dello spettacolo è rappresentato dal corpo del programma

Parole riservate



- Esistono delle parole, dette riservate, a cui non può essere dato il valore di variabile, queste parole sono quelle che compongono le istruzioni ed i tipi di dato.
- Tornando all'esempio del teatro, non possiamo chiamare un personaggio SIPARIO oppure SCENA, esse infatti sono parti specifiche del teatro e non possono essere perciò dei personaggi.

Esempio



- Problema: calcolare il triplo di un numero dato.
- D'ora in poi le parole riservate verranno scritte in maiuscolo.
- E' da ricordare che **alla fine di qualsiasi istruzione bisogna inserire il punto e virgola (;)**. Anche se non è obbligatorio andare a capo dopo di esso, è preferibile farlo per rendere più agevole la lettura e la comprensione del listato.

Commenti al listato



- Per inserire un commento che non interferisca con l'esecuzione del programma bisogna inserire il commento tra parentesi graffe oppure tra parentesi tonde e asterischi
- Es1: {Questo è un commento}
- Es2: (* Anche questo è un commento *)

```
PROGRAM triplo; { Sezione dell' intestazione }
VAR n:INTEGER; { Sezione delle dichiarazioni }
begin { Corpo del programma }
  writeln(' Questo programma calcola il triplo di un
    numero dato ');
  writeln;
  writeln('Introduci un numero intero ');
  readln(n);
  n:=n*3;
  writeln;
  writeln(' Il triplo è ',n);
  writeln;
  writeln(' Premi enter per finire');
  readln;
end.
```





- Nella parte dichiarativa abbiamo comunicato all'elaboratore che `n` è una variabile (**VAR**) che nel corso dell' esecuzione potrà avere soltanto un valore **INTEGER**, cioè di numero intero positivo o negativo.
- La parte esecutiva è introdotta da **BEGIN**.



- Poi troviamo una nuova parola riservata, **READLN** (abbreviazione Read Line = leggi riga).Essa serve a leggere i dati di input.
- La sua sintassi è molto semplice :
 - **READLN(nome variabile da leggere)**



- Inoltre l'istruzione READLN non seguito da alcun parametro, crea un ciclo di attesa che dura finché non viene premuto Invio.
- E' consigliabile metterla alla fine di un listato, prima dell' END quando dobbiamo leggere il risultato di un'elaborazione poiché "congela" lo schermo



- Ad esempio, se eseguiamo il programma precedente senza READLN finale, non potremmo esaminare i risultati della nostra operazione, perché il programma si arresterebbe troppo in fretta.

...

Provare per credere



- Un'altra fondamentale istruzione è **WRITELN** (abbreviazione Write Line = scrivi riga); essa serve a stampare i dati di output sullo schermo.
- La sua sintassi è :
 - **WRITELN('Stringa di testo')**



- Se consideriamo il nostro esempio noteremo che il testo tra apici (') verrà stampato sullo schermo;
- nel caso in cui WRITELN venga usato senza parametri l'elaboratore lascerà una riga vuota nell'elaborazione.



- Un ulteriore esempio sull'uso di WRITELN:
 - **WRITELN(a)**
- In questo caso verrà visualizzato sullo schermo il valore assunto in quel momento dalla variabile a. Dunque, se a vale 3, allora sullo schermo uscirà il numero 3



- E' possibile anche combinare stringhe e variabili, mettendo una virgola tra il secondo apice e la variabile:
 - **Writeln(' Io sono ',A);**



- Alla fine del programma, troviamo **END** seguito da un punto. Esso indica la fine del flusso delle informazioni e il conseguente arresto del programma



DevPascal

- Il devPascal è una applicazione software che mette a disposizione dell'utente un IDE (Integrated Development Environment) per implementare programmi in linguaggio Pascal. Un IDE è un ambiente che propone all'utente molti strumenti di supporto alla fase di implementazione di un programma, strumenti (come editor di testi, compilatore) utili a semplificare il lavoro necessario a costruire un programma.
- Il devPascal è costruito sul compilatore GNUPascal, il compilatore vero e proprio.

Come creare un nuovo Programma

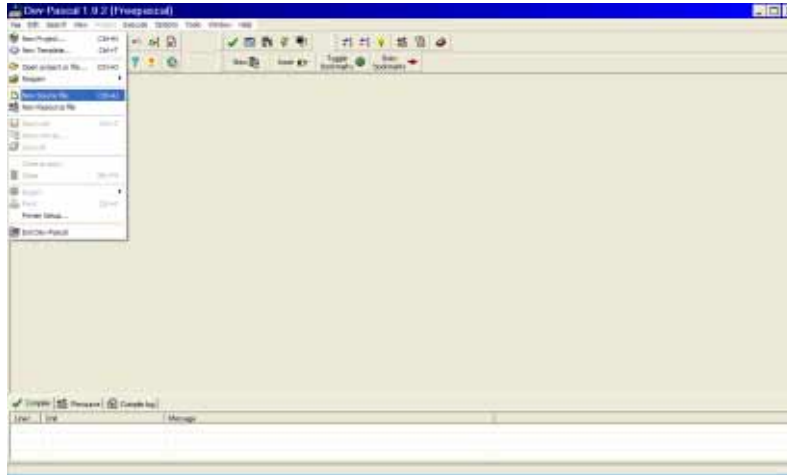


- **Aprire l'ambiente DEV**
 - Sul computer personale: START – Tutti i Programmi – Bloodsheld DevPascal – DevPascal
 - In Laboratorio: START – Programmazione – Bloodsheld DevPascal – DevPascal
- **NOTA: operazione preliminare: creare una cartella in C:\ per contenere i vostri programmi**

L'ambiente DEV



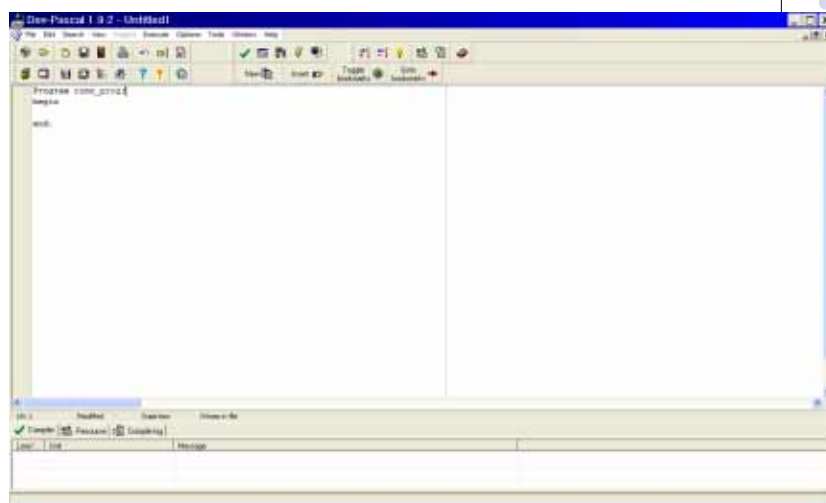
- **Selezionare FILE+New Source File**



Angela Peduto - Informatica generale
A.A. 2005/06

27

- **Eliminare le prime 4 righe e sostituirle con program nome_prog; (o qualunque altro nome volete dare al programma)**



Angela Peduto - Informatica generale
A.A. 2005/06

28

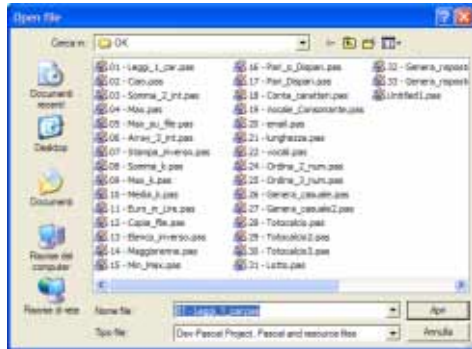
Come aprire un Programma esistente

Aprire l'ambiente DEV

- **Cliccare sull'icona Open Project:**



- **Selezionare il nome del programma e cliccare su apri**

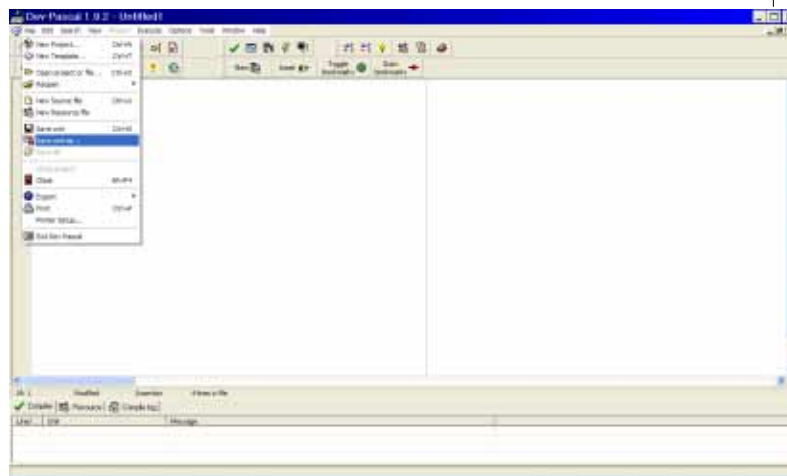


Angela Peduto - Informatica generale
A.A. 2005/06

29

Come salvare un (nuovo) Programma

- **Dall'interno dell'ambiente DevPascal selezionare FILE + Save Unit As ...**



Angela Peduto - Informatica generale
A.A. 2005/06

30

- **Assegnare un nome (ed eventualmente un percorso) al File (es: PROVA.pas) e cliccare su SALVA**



- **Per salvare un programma a cui avete già assegnato il nome basta selezionare l'icona del dischetto.**

Angela Peduto - Informatica generale
A.A. 2005/06

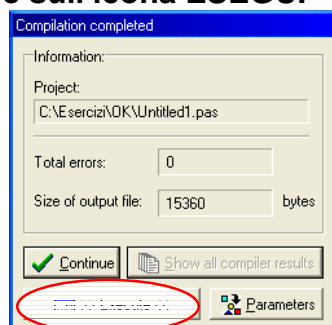
31

Come compilare ed eseguire un Programma

- **Cliccare sull'icona COMPILA**



- Ed aspettare la risposta.
- Se ci sono errori questi vanno corretti e il programma va ricompilato,
- altrimenti **cliccare sull'icona ESEGUI**



Nota: Nella parte inferiore della finestra sono visualizzate informazioni sulla natura e sulla localizzazione di tali errori.

Angela Peduto - Informatica generale
A.A. 2005/06

32



....ora siete pronti a scrivere il vostro
primo programma in Pascal!



```
PROGRAM triplo; { Sezione dell' intestazione }  
VAR n:INTEGER; { Sezione delle dichiarazioni }  
begin { Corpo del programma }  
  writeln(' Questo programma calcola il triplo di un numero  
    dato ');  
  writeln;  
  writeln('Introduci un numero intero ');  
  readln(n);  
  n:=n*3;  
  writeln;  
  writeln(' Il triplo è ',n);  
  writeln;  
  writeln(' Premi enter per finire');  
  readln;  
end.
```