

RELAZIONE DI MATEMATICA: Programma per costruire il triangolo di Pascal

Vogliamo utilizzare la funzione di programmazione di TI-Nspire per costruire un programma che ci mostri il triangolo di Pascal fino alla riga da noi richiesta.

```
pascal
Define LibPriv pascal()=
Prgm
[]
EndPrgm
```

Per prima cosa apriamo una nuova pagina di programmazione, dopodiché scegliamo un nome per il nostro programma (In questo caso, visto l'argomento trattato, **pascal**) Possiamo dividere le tappe del nostro ragionamento per arrivare alla scrittura del programma in quattro fasi: *Ricerca di un "posto" all'interno del triangolo*, *Ricerca di una riga del triangolo*, *Ricerca di tutte le righe a noi necessarie*, *Visualizzazione del risultato*

Ricerca di un "posto" all'interno del triangolo

Questo passaggio fra tutti è il più facile, infatti la macchina ci fornisce già un programma che, inserendo la RIGA ed il POSTO all'interno della riga ci rende il risultato, inseriamo quindi questo comando ($nCr(x,k)$) all'intero del programma.

Ricerca di una riga del triangolo

Per trovare un'intera riga dobbiamo sfruttare una proprietà che noi conosciamo del triangolo di Pascal: il numero dei posti all'interno di una riga è uguale al numero della riga stessa. Utilizziamo quindi un altro comando di TI-Nspire, il comando Seq, che ci fornisca una serie di valori di nCr , il primo sarà il posto 0 della riga x , il secondo il posto 1 così a proseguire fino al posto x della riga x .

$$\text{seq}\{nCr\{x,k\},k,0,x\}$$

Il comando, se scomposto, significa: crea una sequenza di numeri di pascal della riga x (fissa) e di posto K , K varia però da 0 ad X (il massimo che quella riga può raggiungere).

Ricerca di tutte le righe a noi necessarie

Abbiamo adesso scritto la parte di programma che ci fornisce una riga, noi però non vogliamo una sola riga, bensì una serie di righe, a partire dalla riga 0 fino a quella da noi scelta; utilizziamo per fare questo un ciclo For. Il ciclo For è un sottoprogramma che fa compiere alla macchina una serie di cicli di una determinata azione, in questo caso variando ogni volta il valore della riga. Il numero di volte che noi vogliamo far fare alla macchina il calcolo dei valori di una riga è uguale al numero di righe che ci interessano, chiamiamo questo valore n e ordiniamo al programma di compiere il ciclo For tante volte quante n , ogni volta però dovrà aumentare di 1 il valore della riga (Prima la riga 0, poi la riga 1, poi la 2, fino alla riga n).

$$\text{For } x,0,n$$

Diciamo alla macchina che l'operazione che dovrà fare un numero n di volte sarà proprio quella di individuare la riga, andiamo quindi a capo ed inseriamo la formula individuata in precedenza, con il comando EndFor chiudiamo infine il ciclo For.

```

For x,0,n
seq{nCr(x,k),k,0,x}
Endfor

```

Manca però una cosa, x è in funzione di n ma ad n bisogna assegnare ancora il valore, subito all'inizio del programma inseriamo quindi che, perché il programma giri, bisogna inserire un valore di n , in questo modo colui che utilizzerà questo programma potrà scegliere il numero di righe del suo Triangolo di Pascal.

Visualizzazione del risultato

L'ultima cosa rimasta da fare è quella di inserire nel programma un comando che gli imponga di mostrarci il risultato, questo comando è Disp; mettiamo il comando Disp davanti alla formula per individuare la riga, in questo modo la macchina ci fornirà in ordine (Dalla riga 0 alla riga n) tutti i valori calcolati. Salviamo quindi il programma e tentiamo di usarlo; inserisco qui sotto la scrittura completa del programma.

```

* pascal
Define LibPriv pascal(n)=
Prgm
For x,0,n
Disp seq{nCr(x,k),k,0,x}
Endfor
EndPrgm

```

Apriamo una pagina della calcolatrice e utilizziamo il programma per costruire il triangolo di Pascal fino all'ottava riga, per fare questo inseriamo il comando:

Pascal(8)

Come possiamo osservare funziona perfettamente, abbiamo completato il lavoro.

pascal(8)

```

{ 1 }
{ 1,1 }
{ 1,2,1 }
{ 1,3,3,1 }
{ 1,4,6,4,1 }
{ 1,5,10,10,5,1 }
{ 1,6,15,20,15,6,1 }
{ 1,7,21,35,35,21,7,1 }
{ 1,8,28,56,70,56,28,8,1 }

```

Giulio del Corso

Fatto