7a - GLI OPERATORI "DIV" E "MOD"; PARI? DISPARI? DIVISIBILE PER …? DIVISORE DI …?

GLI OPERATORI "DIV" E "MOD" forniscono il quoziente intero e il resto della divisione intera.
a DIV b dà il QUOZIENTE INTERO, a MOD b dà il RESTO.
a, b devono essere numeri, o variabili, di tipo INTERO (con b diverso da 0). Esempi:

30 DIV 7 = 4 ("30 diviso 7" dà 4; poi c'è anche un resto, ma in questo momento non ci interessa)
30 MOD 7 = 2 (il *resto* della divisione "30 diviso 7" è 2)
15 DIV 3 = 5; 15 MOD 3 = 0; 4 DIV 7 = 0; 4 MOD 7 = 4

Per fare esercizi, trascrivi e manda in esecuzione per alcune volte il seguente programmino:

```
program esercizi_sul_div_e_sul_mod;   uses crt;
var a, b, x, y, i : integer;
begin
    clrscr;
    randomize;
    a:=random (100);  b:=random (10)+1;
    write ( a, ' div ', b, ' = ' ); readln (x);   write ( a, ' mod ', b, ' = ' ); readln (y);
    if (x = a div b) and (y = a mod b) then writeln ('OK')
                                       else
                                           begin
                                                writeln ('NO. Risultati esatti: ');
                                                writeln ( a, ' div ', b, ' = ', a div b);
                                                writeln ( a, ' mod ', b, ' = ' , a mod b);
                                           end;
    readln;
end.
```

PARI? DISPARI? DIVISIBILE PER …? DIVISORE DI …?

if a mod b = 0 … significa (come preferisci):

□ **"se a è divisibile per b …"** □ **"se a è multiplo di b …"** □ **"se b è divisore di a …"**

e perciò:

♪ **if x mod 2 = 0 …** significa: **"se x è pari …"**
♫ **if x mod 2 = 1 … (oppure: if x mod 2 <>0)** significa: **"se x è dispari …"**

NOTA: "diverso da" in PASCAL ha come simbolo <>

7b - ANCORA SULLE VARIABILI REAL: CIFRE SIGNIFICATIVE, UNDERFLOW …

Per le variabili di tipo *reale*, destinate ad assumere valori numerici non necessariamente interi, oltre che il discorso sul *range* (= intervallo di variabilità) è importante anche quello relativo al *numero massimo di cifre significative* supportate.
Lo specchietto seguente è tratto dalla documentazione on-line presente su www.freepascal.org:

Type	Range	Significant digits	Size (*byte occupati*)
Real	platform dependant	???	4 or 8
Single	1.5E – 45 .. 3.4E38	7-8	4
Double	5.0E – 324 .. 1.7E308	15-16	8
Extended	1.9E – 4932 .. 1.1E4932	19-20	10

E' evidente che il numero limitato di cifre significative utilizzabili può costringere il computer ad effettuare delle approssimazioni, a seguito delle quali è possibile che si determinino risultati imperfetti.

Tornando poi ad occuparci del *range*, osserviamo che nel caso in cui il valore di una variabile numerica di un determinato "tipo" andasse a superare il massimo del *range* che compete a quel tipo, si avrebbe un *overflow* (traboccamento) e il programma, di norma, si arresterebbe con una segnalazione di errore.

Se al contrario tale valore diventasse (in valore assoluto) troppo piccolo, l'errore sarebbe di *underflow* e purtroppo non verrebbe segnalato: il destino dei numeretti "troppo piccoli" è semplicemente di essere approssimati a 0. E anche da ciò possono derivare alterazioni per i risultati forniti dal programma.
I quali dunque, in determinate situazioni, devono essere interpretati con senso critico, alla luce di quanto detto.
Questo discorso può riguardare ad esempio i programmi per l'approssimazione di π proposti a pagina 23.