
Program prostesi; { όνομα ή επικεφαλίδα ή τίτλος του προγράμματος }
(* αυτό το πρόγραμμα προσθέτει δύο αριθμούς *)
Uses
 Crt; { δήλωση της μονάδας Crt, που περιέχει συναρτήσεις οθόνης }
Var { τμήμα δηλώσεων του προγράμματος }
 a, b, c : Real; { δήλωση τριών αριθμητικών μεταβλητών }
Begin { αρχή του κυρίου προγράμματος, δηλ. των εντολών }
 Read(a, b); { διάβασμα τιμών για δύο μεταβλητές }
 c := a+b; { υπολογισμός του αθροίσματος σε μια τρίτη μεταβλητή }
 Write(c) { εμφάνιση της τιμής της μεταβλητής στην οθόνη }
End. { τέλος του προγράμματος }
(* ο χαρακτήρας ; χρησιμοποιείται για να δηλώσει το τέλος μιας εντολής και έχει λίγες μόνο εξαιρέσεις και το σύμβολο := χρησιμοποιείται για να καταχωρούμε τιμές σε μεταβλητές *)

Program misthos;
(* αυτό το πρόγραμμα κάνει υπολογισμό μισθοδοσίας *)
Uses
 Crt;
Var
 HM, ME, MA, KP, KA : Real;
{ HM είναι το ημερομίσθιο, ME είναι οι μέρες εργασίας, MA είναι οι μικτές αποδοχές, KP είναι οι κρατήσεις και KA είναι οι καθαρές αποδοχές }
Begin
 Read(HM, ME);
 MA := ME*HM;
 KP := 0.20*MA;
 KA := MA-KP;
 Write(MA, KP, KA)
End.

Program fahr_cel;

(* πρόγραμμα μετατροπής θερμοκρασίας Φαρενάιτ σε Κελσίου *)

(* ο τύπος μετατροπής είναι ο εξής : $CEL = ((FAHR-32)*5)/9$ *)

Uses

Crt;

Const

factor = 32.0; { δήλωση μιας σταθεράς }

Var

bath_fahr, bath_cel : Real;

{ bath_fahr είναι οι βαθμοί Φαρενάιτ και bath_cel είναι οι βαθμοί Κελσίου }

Begin

Writeln('Μετατροπή θερμοκρασίας από βαθμούς Φαρενάιτ');

Writeln('σε βαθμούς Κελσίου');

Writeln; { αφήνει μια κενή γραμμή στην οθόνη }

Write('Δώσε τη θερμοκρασία σε βαθμούς Φαρενάιτ :');

Readln(bath_fahr); { διάβασμα της θερμοκρασίας σε Φαρενάιτ }

{ υπολογισμός της θερμοκρασίας σε βαθμούς Κελσίου }

bath_cel := ((bath_fahr-factor)*5.0)/9.0;

Writeln('Αντιστοιχεί σε ', bath_cel, 'βαθμούς Κελσίου');

Writeln; { αφήνει μια κενή γραμμή στην οθόνη }

End.

Program Athroisma;

{ Το πρόγραμμα αυτό υπολογίζει και εκτυπώνει το άθροισμα κάποιων αριθμών, τελειώνει δε όταν το άθροισμά τους ξεπεράσει το 1.000 }

Uses

Crt;

Var

a, s : Real;

Begin

Clrscr;

s := 0; *{ απόδοση αρχικής τιμής στη μεταβλητή }*

Repeat *{ οι παρακάτω εντολές επαναλαμβάνονται μέχρις ότου }*

 Readln(a); *{ η μεταβλητή s γίνει μεγαλύτερη από 1000 }*

 s := s+a

Until s>1000;

Writeln(s);

Repeat until keypressed;

End.

Program Foros;

{ Το πρόγραμμα αυτό υπολογίζει τον φόρο ενός μισθωτού για ένα χρόνο, δίνοντας σαν δεδομένα το ετήσιο εισόδημά του και τον αριθμό των παιδιών του. Ο φόρος υπολογίζεται σαν το ένα τρίτο του εισοδήματος αφού αφαιρέσουμε τις απαλλαγές που είναι η ατομική έκπτωση καθώς και η έκπτωση για το κάθε παιδί }

Uses

Crt;

Const

ekptosh = 500,000; { γενική έκπτωση }
ekptosh_boys = 100,000; { έκπτωση για το κάθε παιδί }

Var

boys, eisodima, forologiteo, foros : Longint;
{ χρησιμοποιούμε τύπο δεδομένων Longint γιατί οι τιμές που θα επεξεργαστούμε είναι μεγαλύτερες από την τιμή 32767 που υποστηρίζει ο τύπος δεδομένων Integer }

Begin

Clrscr;
Writeln('Δώσε το ετήσιο εισόδημα : ');
Readln(eisodima);
Writeln('Δώσε τον αριθμό των παιδιών : ');
Readln(boys);

{ υπολογισμός του φορολογητέου }
forologiteo := eisodima - ekptosh - boys*ekptosh_boys;
foros := forologiteo div 3; *{ υπολογισμός του 1/3 του φορολογητέου }*

{ στην παρακάτω εντολή Writeln συνδυάζουμε εκτύπωση σταθερών τιμών και μεταβλητών, όπου οι σταθερές τιμές, δηλ. τα μηνύματα, είναι πάντα μέσα σε εισαγωγικά και χωρίζονται από τις μεταβλητές με κόμματα }

Writeln('Για εισόδημα : ', eisodima, ' το φορολογητέο είναι : ',
forologiteo, ' και ο φόρος είναι : ', foros);

Repeat until keypressed;

End.

Program CharacterCount;

{ Το πρόγραμμα αυτό μετράει τα γράμματα και τα ψηφία που υπάρχουν σε κάποια δεδομένα που καταχωρούνται από το πληκτρολόγιο και τελειώνει όταν συναντήσει τον χαρακτήρα # }

Uses

Crt;

Var

LetterCount, DigitCount : Integer;

Character : Char;

Begin

Clrscr;

{ όταν χρησιμοποιούμε μεταβλητές σαν μετρητές, πάντα τις μηδενίζουμε στην αρχή του προγράμματος }

LetterCount := 0; *{ μετρητής γραμμάτων }*

DigitCount := 0; *{ μετρητής ψηφίων }*

{ οι παρακάτω εντολές επαναλαμβάνονται όσο ο χαρακτήρας που δίνουμε από το πληκτρολόγιο είναι διάφορος του # και όταν δώσουμε τον χαρακτήρα #, βγαίνουμε από τον βρόχο While }

While Character <> '#' do

Begin

Readln(Character); *{ διάβασμα ενός χαρακτήρα }*

{ αν ο χαρακτήρας είναι μεταξύ των λατινικών A και Z }

If ('A' <= Character) and (Character <= 'Z') then

LetterCount := LetterCount + 1

{ αύξηση του LetterCount }

{ αλλιώς, αν ο χαρακτήρας είναι ένα από τα ψηφία 0 έως 9 }

Else If ('0' <= Character) and (Character <= '9') then

DigitCount := DigitCount + 1

{ αύξηση του DigitCount }

End; *{ end of while }*

Writeln('Βρέθηκαν ', LetterCount, ' γράμματα και ', DigitCount, ' ψηφία');

Writeln;

Repeat until keypressed;

End.

Program DEH;

{ Το πρόγραμμα αυτό υπολογίζει τον λογαριασμό της ΔΕΗ ενός καταναλωτή με δεδομένο ότι για τις πρώτες 100 μονάδες η 1 KWh χρεώνεται 15 δρχ., για τις επόμενες 200 μονάδες χρεώνεται 16 δρχ. και από 201 και πάνω μονάδες χρεώνεται 17 δρχ. Ο ΦΠΑ είναι 18%, το ποσό για την ΕΡΤ είναι 5% και το πάγιο είναι 1.500 δρχ. }

Uses

Crt;

Var

katanalosi, kostos : Real;

Begin

Clrscr;

Writeln('Δώσε την κατανάλωση : ');

Readln(katanalosi);

If katanalosi < 100 then kostos := katanalosi * 15;

If (katanalosi > 100) and (katanalosi < 200) then

kostos := 100 * 15 + (katanalosi - 100) * 16;

If katanalosi > 200 then

kostos := 100 * 15 + 100 * 16 + (katanalosi - 200) * 17;

kostos := kostos * 1.18; { προστίθεται ο ΦΠΑ }

kostos := kostos * 1.05 + 1500; { προστίθεται η ΕΡΤ και το πάγιο }

Writeln('Για τη ΔΕΗ θα πληρώσετε : ', kostos:9:0, ' δρχ.');

Repeat until keypressed;

End.

Program MAX;

{ Το πρόγραμμα αυτό βρίσκει τον μεγαλύτερο από τρεις αριθμούς a, b, c. Χρειάζεται να ορίσουμε μια βοηθητική μεταβλητή max, όπου εκεί θα καταχωρούμε προσωρινά τον μεγαλύτερο αριθμό }

Uses

Crt;

Var

a, b, c, max : Real;

Begin

Clrscr;

Writeln('Δώσε τους τρεις αριθμούς : ');

Readln(a, b, c);

max := a;

If b > max then max := b;

If c > max then max := c;

Writeln('Ο μεγαλύτερος αριθμός είναι ο : ', max);

Repeat until keypressed;

End.

Program Monos_Zygos;

{ Το πρόγραμμα αυτό βρίσκει αν ένας ακέραιος αριθμός είναι μονός ή ζυγός }

Uses

Crt;

Var

k, p, y : Longint;

Begin

Clrscr;

Writeln('Δώσε έναν ακέραιο αριθμό : ');

Readln(k);

p := k div 2; { εύρεση του ημιόλου }

y := k mod 2; { εύρεση του υπολοίπου }

If y = 0 then

 Writeln('Ο αριθμός είναι ζυγός')

Else

 Writeln('Ο αριθμός είναι μονός');

Repeat until keypressed;

End.

Program Misthodosia;

{ Το πρόγραμμα αυτό υπολογίζει τον μισθό ενός εργάτη όταν δίνονται οι μέρες εργασίας του, οι υπερωρίες του σε ώρες και είναι γνωστό ότι οι κρατήσεις είναι το 20% των αποδοχών του, οι υπερωρίες του πληρώνονται με το 20% του ημερομισθίου και ο φόρος είναι 5% }

Uses

Crt;

Var

hmerom, meres_erg, yperories, mikta, krathseis, foros, kathara : Real;

Begin

Clrscr;

Writeln('Δώσε το ημερομίσθιο : ');

Readln(hmerom);

Writeln('Δώσε τις μέρες εργασίας : ');

Readln (meres_erg);

Writeln('Δώσε τις υπερωρίες : ');

Readln(yperories);

{ υπολογισμός μικτών αποδοχών }

mikta := hmerom * meres_erg + yperories * hmerom * 0.2;

krathseis := 0.2 * mikta; { υπολογισμός κρατήσεων }

foros := 0.05 * (mikta - krathseis); { υπολογισμός φόρου }

kathara := mikta - krathseis - foros; { υπολογισμός καθαρών }

Writeln('Οι μικτές αποδοχές είναι : ', mikta,
 ' οι καθαρές αποδοχές είναι : ', kathara,
 ' και οι κρατήσεις είναι : ', krathseis);

Repeat until keypressed;

End.

Program Antistrofh;

{ Το πρόγραμμα αυτό διαβάζει έναν διψήφιο ακέραιο αριθμό και υπολογίζει και εκτυπώνει τον ακέραιο που προκύπτει από την αντιστροφή των ψηφίων του }

Uses

Crt;

Var

x, p, y, x1 : Byte;

Begin

Clrscr;

Writeln('Δώσε έναν διψήφιο ακέραιο αριθμό : ');

Readln(x);

p := x div 10; { εύρεση του πηλίκου }

y := x mod 10; { εύρεση του υπολοίπου }

x1 := y * 10 + p; { σχηματισμός του αντίστροφου αριθμού }

Writeln('Ο νέος αριθμός είναι : ', x1);

Repeat until keypressed;

End.

Program Talhro;

{ Το πρόγραμμα αυτό στρογγυλοποιεί έναν μισθό στο τάληρο, δηλ. αν λήγει σε 0, 1 και 2 τον κάνει να λήγει σε 0, αν λήγει σε 3, 4, 5, 6 ή 7 τον κάνει να λήγει σε 5 και αν λήγει σε 8 ή 9 τον κάνει να λήγει σε 0 αλλά αυξημένο κατά 10 }

Uses

Crt;

Var

m, p, y : Longint;

Begin

Clrscr;

Writeln('Δώσε τον μισθό : ');

Readln(m);

p := m div 10; { εύρεση του πηλίκου }

y := m mod 10; { εύρεση του υπολοίπου }

{ το y είναι στην ουσία το τελευταίο ψηφίο του μισθού και το p είναι όλα τα υπόλοιπα ψηφία του μισθού χωρίς το y, δηλ. αν ο μισθός είναι 156967, το y θα είναι ίσο με 7 και το p θα είναι ίσο με 15696 }

If y <= 2 then m := p * 10;

*{ 15696 * 10 = 156960 }*

If (y > 2) and (y < 8) then m := p * 10 + 5;

*{ 15696 * 10 + 5 = 156965 }*

If y >= 8 then m := p * 10 + 10;

*{ 15696 * 10 + 10 = 156970 }*

Writeln('Ο στρογγυλοποιημένος μισθός είναι : ', m);

Repeat until keypressed;

End.