«Ηλεκτρονικοί Υπολογιστές»
7ο εξάμηνο

«Τεχνικές και μέθοδοι προγραμματισμού με τη γλώσσα προγραμματισμού C++ Θεωρία και ασκήσεις»
2011

Ζαχαρούλα Σ.Ανδρεοπούλου
Επίκουρος Καθηγήτρια ΑΠΘ

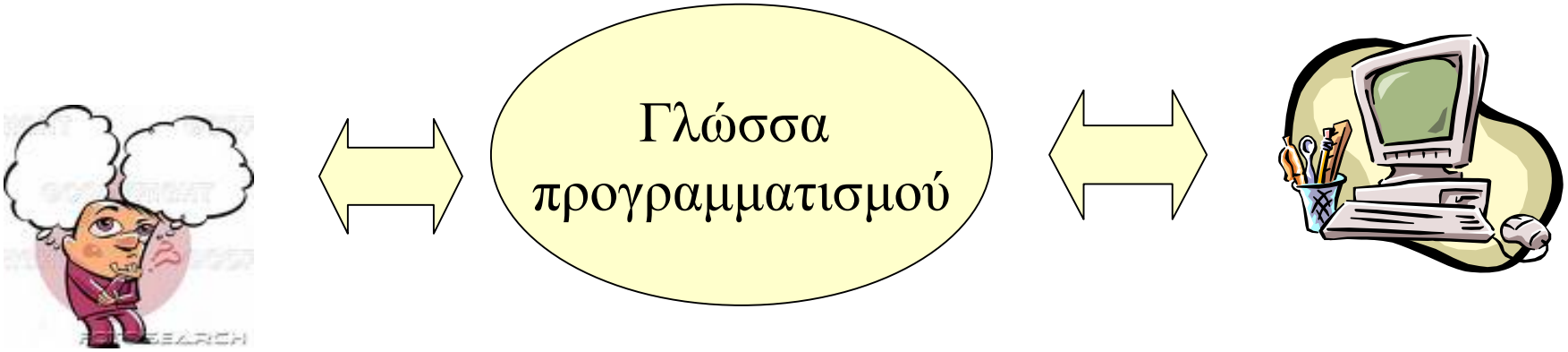
Γλώσσα προγραμματισμού

- Για να λύσει ένα πρόβλημα ο Η/Υ ακολουθεί συγκεκριμένες εντολές.
- Το σύνολο των εντολών αποτελούν ένα πρόγραμμα
- Το σύνολο των κανόνων στα πλαίσια των οποίων γράφουμε τις εντολές =ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Αν γνωρίζουμε τον αλγόριθμο λύσης του προβλήματος,

Τον μετατρέπουμε σε πρόγραμμα για τον Η/Υ

Σε μια γλώσσα προγραμματισμού



Κατηγορίες γλωσσών προγραμματισμού

- Γλώσσες υψηλού επιπέδου-ανώτερες γλώσσες προγραμματισμού (high level languages)
- Γλώσσες μηχανής (machine languages)
- Assembly languages (συμβολικές γλώσσες προγραμματισμού)

Γραφή προγράμματος στη C++

- Το αλφάβητο της C++:
  - αγγλικό αλφάβητο a.b.c.d.e.f.g.h.I.j.k.l.m.n.o.p.q.r.s.t.y.v.w
    - (κεφαλαία και μικρά)
  - 10 αριθμητικά ψηφία 1,2,3,4,5,6,7,8,9,0
  - Ειδικά σύμβολα ! # $ % & * ( ) + >< ' ;
  - Οι ελληνικοί χαρακτήρες μπορεί να είναι μόνο εισερχόμενα δεδομένα

Από το Πηγαίο πρόγραμμα (source)

- Προ-επεξεργαστής (διαγραφή σχολίων και προσθήκη απαραίτητων προγραμμάτων)
- Μεταγλωττιστής (συμβολική γλώσσα)
- Σύμβολο- μεταφραστής (γλώσσα μηχανής)
- Δια-συνδέτης (προσθήκες για να γίνει εκτελέσιμο
  - Σύνδεση με βιβλιοθήκες, εκχωρήσεις μνήμης, κ.λπ.
  - Executable EXE

Λάθη μεταγλώτισσης

ΚΑΝΟΝΕΣ ΓΡΑΦΗΣ

- Κάθε γραμμή τελειώνει πάντοτε με το σύμβολο ;
- Κάθε γραμμή στο πρόγραμμα
  - δεν ξεπερνά τους 248 χαρακτήρες
  - Πρακτικά τους 80

Δομή προγράμματος C++

Κάθε πρόγραμμα C++ αποτελείται από 5 τμήματα

- Αρχεία-κεφαλίδες → # include <iostream.h>
- Δηλώσεις συναρτήσεων → Void main ( )
  {
- Δηλώσεις μεταβλητών → int a;
  float b;
- Αρχικές τιμές – υπολογισμοί → a=12;
- Εκτυπώσεις → Cin >>a;
  Cout <<a;
  }

include <iostream.h>
header file- αρχείο κεφαλίδα

- Δηλώνει την ενσωμάτωση αυτού του αρχείου από την βιβλιοθήκη της C++
- Αρχίζει πάντοτε με #
- Κάθε αρχείο κεφαλίδα δηλώνεται σε μια γραμμή
- ΔΕΝ υπάρχει ; στο τέλος

ΣΥΝΑΡΤΗΣΕΙΣ

- { αρχίζει η συνάρτηση
- } τελειώνει η συνάρτηση
- Void δεν επιστρέφει τιμές
- Main η κύρια συνάρτηση του προγράμματος

ΔΕΔΟΜΕΝΑ στη C++

- Αριθμοί (numbers)
- Χαρακτήρες (characters)
- Συμβολοσειρές (Strings)

ΑΡΙΘΜΟΙ

Σημαντικά ψηφία & μέγεθος

- **Ακέραιοι** -32768 έως και +32767
  - 5 σημαντικά ψηφία (2 bytes)
  - 10 σημαντικά ψηφία (4 bytes)
- **Πραγματικοί**
  - Θετικοί 3.4×10^{-38} έως και $3.4 \times 10^{+38}$
  - Αρνητικοί $-3.4 \times 10^{+38}$ έως και -3.4×10^{-38}

  - 7 σημαντικά ψηφία (4 bytes)

ΣΗΜΑΝΤΙΚΑ ΨΗΦΙΑ- όλα τα ψηφία εκτός από 0 στην αρχή και στο τέλος του αριθμού(δεκαδικού)

ΑΡΙΘΜΟΙ ΔΙΠΛΗΣ ΑΚΡΙΒΕΙΑΣ

- 15 σημαντικά ψηφία
  - 8 bytes

Αρνητικοί διπλής ακρίβειας

$-1.7 \times 10^{+308}$ έως και -1.7×10^{-308}

Θετικοί διπλής ακρίβειας

$+1.7 \times 10^{-308}$ έως και $+1.7 \times 10^{+308}$

•Μεγάλοι αριθμοί διπλής ακρίβειας

- ΕΚΘΕΤΙΚΟΙ ΑΡΙΘΜΟΙ
  - Οι αριθμοί διπλής ακρίβειας γράφονται και σε εκθετική μορφή
    - $12.35 \times 10^2 \longrightarrow$ 12.35e2
    - UNDERFLOW
    - OVERFLOW

ΧΑΡΑΚΤΗΡΕΣ

- Οποιοσδήποτε χαρακτήρας του Η/Υ
- Γράφεται σε μονά εισαγωγικά
  - Π.χ. ‘α’
  - Π.χ. ‘?’
  - Π.χ. ‘5’

ΣΥΜΒΟΛΟΣΕΙΡΕΣ

- Ακολουθία χαρακτήρων σε διπλά εισαγωγικά=συμβολοσειρά
- Περιέχει κανένα, ένα ή πολλούς χαρακτήρες
- Η σειρά των χαρακτήρων είναι συγκεκριμένη
  - Π.χ. «ΒΑΣΙΚΟΣ»
  - Π.χ. «235»
  - Π.χ. «K32/221-BA»
  - Π.χ. «»
  - Π.χ. «a»

Βρείτε το πλήθος των σημαντικών ψηφίων των παρακάτω αριθμών…

1. 620300 → **6**
2. 003420 → **4**
3. 0.245 → **3**
4. 0.0204050 → **5….20405 x 10^{-6}**
5. 0.2008010 → **6**

Σωστός ή λάθος αριθμός για τη C++?

1. 126 👍
2. 12,78 👎
3. 0.008910001 👍
4. 245L45 👎
5. -027645e4 👍
6. +23.76e408 👎

Μεταβλητές στη C++

ΚΑΤΗΓΟΡΙΕΣ ΜΕΤΑΒΛΗΤΩΝ

- Απλές μεταβλητές αριθμών ή αριθμητικές μεταβλητές
- Μεταβλητές χαρακτήρων
- Μεταβλητές με δείκτη (Πινάκων και συμβολοσειρών)

ΟΝΟΜΑΤΑ ΜΕΤΑΒΛΗΤΩΝ

✓ Συνδυασμός γραμμάτων (Κεφαλαίων ή πεζών) του αγγλικού αλφαβήτου ή ψηφίων (0,1,2,…,9) και του _

- Πρέπει να αρχίζουν με γράμμα ή με το _
- Να μην περιέχουν ειδικά σύμβολα (! # $ % & * ( ) + >< ' ;)
- Να έχουν ορισμένο πλήθος χαρακτήρων (μέχρι 32 συνήθως)
- Δεν επιτρέπεται να περιέχουν κανένα γράμμα του ελληνικού αλφαβήτου

Σωστή ή λάθος μεταβλητή για τη C++?

1. x45R
2. _bHr
3. 2ax
4. zxψ
5. a%f
6. a_b

Η C++ κάνει διάκριση μεταξύ πεζών και κεφαλαίων. Π.χ AB_1 , ab_1 , Ab_1 δεν είναι ίδιες αλλά τρείς διαφορετικές μεταβλητές

ΔΗΛΩΤΙΚΕΣ ΕΝΤΟΛΕΣ ΜΕΤΑΒΛΗΤΩΝ

- Για ακέραιους Int (Integer)

  Π.χ: Int x,par,xab1;

- Για μεγάλους ακέραιους long int ή απλά long

  Π.χ: long int x2,name;

  Π.χ: long x2,name;

- Για ακεραίους χωρίς πρόσημο Unsigned

  Π.χ: Unsigned int k,l,dl;

  Π.χ: unsigned long xl,sal,net;

ΔΗΛΩΤΙΚΕΣ ΕΝΤΟΛΕΣ ΜΕΤΑΒΛΗΤΩΝ

- Για Πραγματικούς αριθμούς float

  Π.χ: float x,par,xab1;

- Για μεταβλητές διπλής ακρίβειας double

  Π.χ: double int x2,name;

- Για μεγάλες μεταβλητές διπλής ακρίβειας long double
- Για μεταβλητές οι τιμές των οποίων θα είναι χαρακτήρες char

ΑΡΧΙΚΕΣ ΤΙΜΕΣ ΜΕΤΑΒΛΗΤΩΝ

- Η αρχική τιμή μιας μεταβλητής δηλώνεται με 2 τρόπους μέσα στο πρόγραμμα

1ος Τρόπος

```
Int a, b, c2a ;
Float fl, dd2 ;
Char cl, c2 ;
```

Δήλωση μεταβλητών

```
a = 2;
b = -1245;
c2a = 23000;
fl = 2345.7987;
dd2 = 23.89e-12;
cl = 'a';
c2 = 'A';
```

Ορισμός αρχικών τιμών μεταβλητών

2ος Τρόπος

```
Int a=2, b=-1245, c2a= 23000;
Float fl= 2345.7987, dd2= 23.89e-12 ;
Char cl= 'a', c2 = 'A';
```

Ταυτόχρονη δήλωση μεταβλητών και ορισμός της αρχικής τιμής τους

ΑΡΙΘΜΗΤΙΚΕΣ ΠΡΑΞΕΙΣ ΣΤΗ ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ C++

ΕΚΤΕΛΕΣΗ ΑΡΙΘΜΗΤΙΚΩΝ ΠΡΑΞΕΩΝ

Χρησιμοποιούμε σταθερές και μεταβλητές σε 4 πράξεις

ΠΡΑΞΗ	ΣΥΜΒΟΛΟ
Πρόσθεση	+
Αφαίρεση	-
Πολλαπλασιασμός	*
Διαίρεση	/

Π.χ: a+3.5*(a+b)-x+y/a

Για **<u>Ύψωση αριθμού σε δύναμη</u>** η C++ χρησιμοποιεί τη συνάρτηση βιβλιοθήκης **pow(power)**.

Π.χ: pow(x,y) θα μας δώσει x^y

Για **<u>Υπόλοιπο διαίρεσης</u>** μεταξύ ακεραίων **α%β**

Π.χ. αν α=15 και β=4 α%β δίνει αποτέλεσμα 3 γιατι

Σειρά εκτέλεσης αριθμητικών πράξεων

Βαθμός προτεραιότητας	Πράξη
1η	αλλαγή πρόσημου
2η	δυνάμεις
3η	πολ/σμοί, διαιρέσεις
4η	προσθέσεις,αφαιρέσεις

<u>ΚΑΝΟΝΕΣ</u>

ΚΑΝΟΝΑΣ 1.Όταν έχουμε συνεχόμενες πράξεις ίδιας προτεραιότητας τότε η σειρά εκτέλεσης είναι από αριστερά προς τα δεξιά

π.χ a/b/c θα γίνει πρώτα κ=α/ b και κ/c

ΚΑΝΟΝΑΣ 2.

-Γίνονται πρώτα οι πράξεις εντός των παρενθέσεων μέχρι να δημιουργηθεί παράσταση χωρίς παρενθέσεις.

-Οι πράξεις αρχίζουν από τις εσωτερικές προς τις εξωτερικές παρενθέσεις

a+b*(x+y)-a/(x-y)+pow(b,(-I+K)) >>> a+b*x1-a/x2+pow(b,j)

ΚΑΝΟΝΑΣ 3. Δεν μπορούμε να βάλουμε δύο σύμβολα πράξεων το ένα δίπλα στο άλλο.

ΚΑΝΟΝΑΣ 4. Πρέπει να τα χωρίσουμε με παρενθέσεις

X+a/-b Είναι λάθος

X+a/(-b) Είναι σωστό

!!!!!!Όπου αμφιβάλετε χρησιμοποιείστε παρενθέσεις

ΚΑΝΟΝΑΣ 5.

Ύψωση βάσης χ σε δύναμη ψ

X^{Ψ} Όπου ψ= αλγεβρική παράσταση

ΔΕΝ ισχύει όταν

- $X<0$ και ψ=πραγματικός
- $X=0$ και $\psi<0$

ΚΑΝΟΝΑΣ 6. Διαίρεση με το 0

Δεν μπορούμε να διαιρέσουμε μια σταθερά η μια μεταβλητή

-με το 0

- με μεταβλητή που παίρνει την τιμή 0

ΕΚΤΕΛΕΣΗ ΑΡΙΘΜΗΤΙΚΩΝ ΠΡΑΞΕΩΝ

Παραδείγματα αριθμητικών πράξεων στη C++

- *Να μετατραπούν σε παραστάσεις C++ οι παρακάτω αλγεβρικές παραστάσεις:*

1) $b = (a+2x)\ (a-2x)$ → 1) `b = (a+2*x) * (a-2*x)`

2) $z = 1/x + b^2/4w^3$ → 2) `z = 1.0/x + pow(b,2) / (4*pow(w,3));`

3) $\rho = 27{,}8\ x + ab + km$ → 3) `r = 27.8 *x+ a*b+ k*m;`

4) $x = a\ (\ a^2-b^2\ )\ /\ (b-1)$ → 4) `x = a*(pow(a,2) - pow(b,2))/(b-1);`

5) $\mu = k^2 + (m-1)^2 - 3km/(i-k)^2$ → 5) `m = pow(k,2) +pow(m-1,2) - 3*k*m/pow(i-k,2);`

ΕΚΤΕΛΕΣΗ ΑΡΙΘΜΗΤΙΚΩΝ ΠΡΑΞΕΩΝ

Παραδείγματα αριθμητικών πράξεων στη C++

- *Να μετατραπούν σε παραστάσεις C++ οι παρακάτω αλγεβρικές παραστάσεις:*

1) $\mu = \kappa\,(\kappa - 1{,}2^{n})$ → 1) m = k * (k – pow (1.2, n));

2) $\Omega = -\dfrac{1}{1-x} + \dfrac{2}{\sqrt{x}} + 1$ → 2) W = -1/(1-x) +2 / sqrt (x)+1;

3) $\Delta = \dfrac{3x^2 - 5x}{\sqrt{(\beta^2 - 2a)}}$ → 3) D = 3*pow(x,2) – 5*x / sqrt (pow(b,2) – 2 * a);

ΕΚΤΕΛΕΣΗ ΑΡΙΘΜΗΤΙΚΩΝ ΠΡΑΞΕΩΝ

Άσκηση

- *Να γραφεί κώδικας που να υπολογίζει τον όγκο σφαίρας.*

Δίνεται ο τύπος $V = \frac{4}{3} \pi R^3$

ΕΝΤΟΛΕΣ ΑΝΤΙΚΑΤΑΣΤΑΣΗΣ ΣΤΗ C++

ΕΝΤΟΛΕΣ ΑΝΤΙΚΑΤΑΣΤΑΣΗΣ

Η γενική μορφή της είναι: a = b

✓ Όπου: a = μεταβλητή

b = μεταβλητή ή σταθερά ή αλγεβρική παράσταση

Το περιεχόμενο της b αντικαθιστά το περιεχόμενο της a

Με εντολές αυτής της κατηγορίας μπορούμε:

- Να δώσουμε μια τιμή (αριθμό ακέραιο ή πραγματικό) σε μια μεταβλητή. Π.χ: k=345; *Δίνουμε την τιμή 345 στη μεταβλητή k.*
- Να μεταφέρουμε την τιμή μιας μεταβλητής σε μια άλλη μεταβλητη. Π.χ: k= 345;

a=432.19;

m=k;

b=a;

Μετά την εκτέλεση αυτών των εντολών m=345 & b=432.19

ΕΝΤΟΛΕΣ ΑΝΤΙΚΑΤΑΣΤΑΣΗΣ

✓ **Με εντολές αυτής της κατηγορίας μπορούμε:**

- Να κάνουμε διάφορες πράξεις μεταξύ μεταβλητών και σταθερών και το αποτέλεσμα να το βάλουμε σε μια μεταβλητή. Π.χ:

*x1=(a+b)*c-a/b+4.5;*

*z=x1*2+4;*

- *Εννοείται ότι οι μεταβλητές a,b έχουν πάρει τιμές προηγουμένως.Στη συνέχεια γίνονται οι πράξεις και η z παίρνει το αποτέλεσμα που προκύπτει.*
- *Γενικά μία μεταβλητή μπορεί να αλλάζει συνεχώς τιμές μέσα στο ίδιο πρόγραμμα με τη χρήση της εντολής αντικατάστασης.*

ΕΝΤΟΛΕΣ ΑΝΤΙΚΑΤΑΣΤΑΣΗΣ

Γενικά θα μπορούσαμε να γράψουμε:

Νέα τιμή = Παλαιά Τιμή + Τιμή

- *Εντολές αυτής της μορφής είναι χρήσιμες στον προγραμματισμό* Π.χ:

a = a + b

i = i +1

✓Η πρώτη αυξάνει την τρέχουσα τιμή του a κατά b κάθε φορά που εκτελείται

✓Η δεύτερη αυξάνει την τρέχουσα τιμή του i κατά 1.

ΣΥΝΑΡΤΗΣΕΙΣ ΒΙΒΛΙΟΘΗΚΗΣ

Απόλυτος τιμή	fabs( )
Τετρ.ρίζα	sqrt ( )
Δύναμη	pow (x,y)
Λογάριθμος	log ( )
Ημίτονο (γωνίας σε ακτίνια)	sin ( )
Συνημίτονο ……	cos ( )
Εφαπτομένη	tan ( )

Συνάρτηση exit (ο)

Η εκτέλεση του προγράμματος σταματά

-όταν φτάσουμε στην τελευταία } και το πρόγραμμα έχει φτάσει στο τέλος του

- όταν βρούμε μια εντολή exit (0)

Τοποθετείται οπουδήποτε στο πρόγραμμα

Όταν θέλουμε να σταματήσουμε την εκτέλεση του προγράμματος ύστερα από την εκπλήρωση μιας συνθήκης

```
Z= a+a/m-b;

if Z=0   exit(0);
```

Σχόλια στο πρόγραμμα

Γραπτές σημειώσεις που αφορούν το πρόγραμμα και διευκολύνουν εμάς ή άλλους που διαβάζουν το πρόγραμμα.

Δεν περιέχουν εντολές & αγνοούνται από τον ΗΥ κατά την μετάφραση του προγράμματος. Εμφανίζονται όμως στην εκτύπωση της λίστας του προγράμματος ή στην οθόνη. Περιέχουν οδηγίες, επεξηγησεις, πληροφορίες,

Π.χ. τι κάνει το πρόγραμμα, τι σημαίνουν οι μεταβλητές, τι κάνει μια υπορουτινα, σχόλια για τον αλγόριθμο, αλλά και πότε γράφτηκε, από ποιον,

Κάθε σχόλιο αρχίζει με /* και τελειώνει με */

π.χ. B=20.0; /* B einai to baros */

ΚΑΙ Σύντομα σχόλια σε μια γραμμή

π.χ. Int k=200; // k= kefalaio

Εντολές εκτύπωσης
στη
Γλώσσα προγραμματισμού C++

printf

ΕΝΤΟΛΗ ΕΚΤΥΠΩΣΗΣ Printf

!!!! Συντάσσεται με την κεφαλίδα # include <stdio.h>

Με την εντολή **printf** μπορούμε να

- Τυπώσουμε σταθερές, δηλ. κείμενα,μηνύματα, αριθμούς, παραστάσεις, χαρακτήρες)
- Τυπώσουμε μεταβλητές (τις τιμές τους)
- Τυπώσουμε αποτελέσματα πράξεων (επιμέρους & τελικά)

ΕΝΤΟΛΗ ΕΚΤΥΠΩΣΗΣ Printf

Η γενική μορφή της εντολής είναι:

printf (“……..”, a1,a2,a3,……ak)

Φόρμα εκτύπωσης

ΠΩΣ τυπώνουμε

ΤΙ τυπώνουμε
(σταθερές, μεταβλητες, παραστασεις C++, συναρτήσεις)

Στη <u>φόρμα εκτύπωσης</u> περιλαμβάνονται

\n αλλαγή γραμμής (χαρακτήρας διαφυγής)

% wd εκτύπωση ακεραίου w θέσεων

% w.qf εκτύπωση πραγματικού w θέσεων με q δεκαδικά ψηφία

% s εκτύπωση σταθερής συμβολοσειράς

% c εκτύπωση χαρακτήρα

χαρακτήρας διαφυγής

- **\n**

Χρησιμοποιείται μέσα στην printf για αλλαγή γραμμής & καθαρή εκτύπωση

Η c++ ερμηνεύει τους 2 χαρακτήρες σαν 1, (τον ***χαρακτηρα διαφυγης)***

π.χ. Printf (“\nABC \nXYZ”);
Θα τυπώσει

ABC
XYZ

ΕΝΤΟΛΗ ΕΚΤΥΠΩΣΗΣ Printf

ΠΑΡΑΔΕΙΓΜΑ ΕΚΤΥΠΩΣΗΣ

Printf (“\naaaaaaa”);

Printf(“\nABC”);

Printf(“XYZ”);

Printf(“\naaaaaaa”); *Θα τυπώσει*

aaaaaaa

ABCXYZ

aaaaaaa

ΕΚΤΥΠΩΣΗ με την Printf

- *Δώστε τις κατάλληλες printf για να εκτυπωθούν οι παρακάτω μεταβλητές:* a=2453, b=365.5234, c="Forest informatics",

1) Printf ("%4d",a);

2) Printf ("%3.4f",b);

3) Printf ("%s",c);

ΕΚΤΥΠΩΣΗ με την Printf

- *Δώστε τις κατάλληλες printf για να εκτυπωθούν οι παρακάτω μεταβλητές:* d='g', e=18.0920 και f=1.0

1) Printf ("%c",d);

2) Printf ("%2.3f",e);

3) Printf ("%1.1f",f);

ΕΚΤΥΠΩΣΗ με την Printf

- *αν:* a=1821, b=1940, d="ΌΧΙ" δώστε την κατάλληλη printf για να εκτυπωθεί η παρακάτω φράση:

Η επανάσταση του 1821 και το ΟΧΙ του 1940

(2 κενές γραμμές)

είναι οι δύο εθνικές μας γιορτές

Printf (" Η επανάσταση του %4d και το %s του %4d \n\n\n είναι οι δύο εθνικές μας γιορτές",a,d,b);

ΕΝΤΟΛΗ ΕΚΤΥΠΩΣΗΣ Printf

Να τυπωθεί ο τίτλος:
«ΑΘΗΝΑ-ΘΕΣΣΑΛΟΝΙΚΗ =504 χιλιόμετρα»

```
/* ΕΚΤΥΠΩΣΗ ΣΤΑΘΕΡΩΝ ΣΥΜΒΟΛΟΣΕΙΡΩΝ */

 # include <iostream.h>

# include <stdio.h>
Void main ( )
{
 int b;
b=504;
Printf ("\n \n ΑΘΗΝΑ-ΘΕΣΣΑΛΟΝΙΚΗ=%3d χιλιόμετρα", b);
}
exit (0);
```

Ένας ακέραιος 3 θέσεων

Πρόγραμμα για το εμβαδόν τριγώνου

```
/* υπολογισμός εμβαδού τριγώνου*/
/* η πρώτη άσκηση στη C++   */
 # include <iostream.h>
  #  include <stdio.h>
Void main ( )
{   int base, ypsos;  // μεταβλητές για τη βάση και το ύψος
    float embadon;  // το εμβαδόν είναι πραγματικός
  base =9,  ypsos = 5; // δίνουμε αρχικές τιμές μέσα στο πρόγραμμα
embadon = base * ypsos /2.0;
Printf ( "\n \n  βάση=  %d μέτρα \n ύψος=  %d μέτρα" , base, ypsos);
Printf("\n \n εμβαδον = %f τετρ.μέτρα" , embadon); }
Exit (0);
```

ΕΚΤΥΠΩΣΗ ΧΑΡΑΚΤΗΡΩΝ

Να γραφεί πρόγραμμα που τυπώνει τους χαρακτήρες a και M

```
/* εκτύπωση συγκεκριμένων χαρακτήρων, 2η άσκηση στη c++ */
# include <iostream.h>
# include <stdio.h>
Void main ( )
{   char  c1, c2;
c1='a';
C2='M';
Printf( "\n \n ο χαρακτηρας c1= %c" , c1);
Printf ( "\n \n ο χαρακτηρας c2= %c ", c2);
Exit (0) ; }
```

Συνδυασμός printf scanf

Ο συνδυασμός των εντολών printf και scanf χρησιμοποιείται για να δώσουμε οδηγίες στον χρήστη του προγράμματος για τις τιμές που θα εισαχθούν μέσω της εντολής scanf

Printf(“\n\n δώσε το πλήθος των μετρήσεων n”);

Scanf(%d, &n);

Printf (“\n\n δώσε την ακτίνα του κορμοτεμαχίου R σε μέτρα”);

Scanf (%f, &R);

Εντολή εισόδου δεδομένων στη Γλώσσα προγραμματισμού C++ scanf

Είσοδος δεδομένων με την εντολή scanf

Μπορούμε να δίνουμε τιμές από το πληκτρολόγιο, δηλαδή εξωτερικά (εκτός προγράμματος), ενώ το πρόγραμμα βρίσκεται σε εξέλιξη.

Scanf ( );

•Περιγράφουμε τη φόρμα δεδομένων που θα δεχτεί η εντολή scanf

•Γράφουμε τα ονόματα των μεταβλητών οι οποίες θα ΔΕΧΘΟΥΝ τα δεδομένα/τιμές.

•Για κάθε scanf απαιτείται να προηγείται και μια εντολή printf που θα τυπώνει οδηγίες για το χρήστη

Εντολή εισόδου δεδομένων scanf

Η γενική μορφή της εντολής είναι:

scanf (“%d %f %c”, &x1, &x2, &x3)

Φόρμα ανάγνωσης

Μεταβλητές που παίρνουν τιμές εξωτερικά

Στη <u>φόρμα ανάγνωσης</u> περιλαμβάνονται

%d σημαίνει ότι θα εισάγουμε ακέραιο αριθμό

%f σημαίνει ότι θα εισάγουμε πραγματικό αριθμό

% c σημαίνει ότι θα εισάγουμε 1 χαρακτήρα

Ο χαρακτήρας «&»

Μπροστά από κάθε μεταβλητή που θα ΔΕΧΘΕΙ τιμή βάζουμε το χαρακτήρα &

Printf (“\n\n Dwse tis times a1,a2,a3:”);

Scanf(“%d,%d,%d”,&a1,&a2,&a3);

Θα δώσουμε τα δεδομένα

Π.χ. 55,60,75

- **Printf (“\n\n Dwse tis times x1,a2,k :”);**
- **Scanf(“%d %f %c”, &x1, &a2, &k);**
- **Θα δώσουμε τα δεδομένα**
- **Π.χ. 55 60.35 h**

Να γραφεί πρόγραμμα που να μετατρέπει τις μοίρες σε ακτίνια τύπος α=π*m/180

```
/* Μετατροπή μοιρών σε ακτίνια*/
 #  include <iostream.h>
#  include <stdio.h>
Void main ( )
{   float m, a, pi;  // μοίρες, ακτίνια, π
   pi=3.14;
Printf ( "\n \n  dwse tis moires:");
Scanf (" %f ", &m);
a=pi*m/180;
Printf ( "\n \n  aktinia = %f ", a); }
Exit (0);
```

Θα εμφανιστεί στην οθόνη το εξής:

Dwse tis moires _ (Γράφουμε την τιμή)

Aktinia=###

Να γραφεί πρόγραμμα που να μετατρέπει τις μοίρες σε ακτίνια τύπος α=πm/180

ΕΝΑΛΛΑΚΤΙΚΗ ΕΚΤΥΠΩΣΗ

Printf ("\n\n Oi %f moires einai %f aktinia", m, a);

Θα εμφανιστεί στην οθόνη το εξής:

Oi ### moires einai ### aktinia

Να γραφεί πρόγραμμα που να υπολογίζει τον όγκο παραλληλεπιπέδου V=abc

Δίδεται εξωτερικά το ύψος, πλάτος και μήκος (ακέραιοι)

```
/* Υπολογισμός όγκου παραλληλεπιπέδου*/
 # include <iostream.h>
#  include <stdio.h>
Void main ( )
{
Int ypsos, platos, mikos, V; //Δίνω τις μεταβλητές
Prinf (“dwse to ypsos to platos kai to mikos se ekatosta”);
Scanf (“%d %d %d”, &ypsos, &platos, &mikos);
V= ypsos*platos*mikos;
Printf(“\n\n O ogkos einai %d  kybika ekatosta”, V);
}
Exit(0);
```

Δίδεται σφαίρα ακτίνας r να υπολογιστεί ο όγκος της. $V=4\pi r^3/3$

```
/* Υπολογισμός όγκου σφαίρας*/

# include <iostream.h>
 #  include <stdio.h>
#include<math.h>
Void main ( )
{
Int r; //Ακτίνα σφαίρας
Float pi, V ;// π και όγκος
pi = 3.14;
Printf (“\n dwse tin aktina se xiliosta”);
Scanf (“%d”, &r);
V= 4*pi*pow(r,3)/3;
Printf(“\n\n O ogkos einai %f kybika xiliosta”, V);
}
Exit(0);
```

Επαναληπτικές εντολές if & for στη Γλώσσα προγραμματισμού C++

Στον προγραμματισμό θέλουμε μια ή περισσότερες εντολές να εκτελούνται όχι 1 αλλά πολλές φορές.

Για το λόγο αυτό υπάρχουν οι επαναληπτικές εντολές

For loops	βρόχος for
While loops	βρόχος while
Do while	βρόχος do while

Για <u>συγκεκριμένο αριθμό επαναλήψεων</u> χρησιμοποιούμε το βρόχο FOR!!!!!

ΜΗ ΣΕΙΡΙΑΚΗ ΕΚΤΕΛΕΣΗ ΕΝΤΟΛΩΝ

Εντολές if

Μια ή περισσότερες εντολές εκτελούνται μόνο όταν ισχύει μια συνθήκη ή συνθήκες.

Χρησιμοποιούνται εντολές σύγκρισης

- Μεταβλητή και αριθμός
- Μεταβλητή με μεταβλητή
- Μεταβλητή με αλγεβρική παράσταση

Και αποφασίζουμε ποια ή ποιες εντολές θα εκτελεστούν στη συνεχεια

Απλή και σύνθετη if

- **ΑΠΛΗ if**
- If (συνθήκη) ..... ;
- If (συνθήκη)
  ..... ;

- **ΣΥΝΘΕΤΗ if**
  - If (συνθήκη)

{.....;
.....; Περιοχή ελέγχου της if
.....; Σώμα της if
} (προσοχή!!!!! δεν χρειάζεται ;)

Τελεστές συνθηκών

Για τις Εντολές IF-τελεστές σύγκρισης

Ισότητα	If (A= = 5) {……. }
Μικρότερο	If (k < 9) {……. }
Μεγαλύτερο	If (2 > 10) {……. }
Μικρότερο ή ίσον	If (a2 < = 1) {……. }
Μεγαλύτερο ή ίσον	If (k4 > = 0) {……. }
Διάφορο (δεν είναι)	If (k **!=** 0) {……. }

Όταν ισχύει η συνθήκη (είναι αληθής) <u>τότε</u> εκτελούμε τις εντολές που ακολουθούν μέσα σε άγκιστρα

- Αν η περιοχή ελέγχου της if είναι μόνο μια εντολή δεν χρειάζονται άγκιστρα { }

- Στις εντολές if η γλώσσα προγραμματισμού C++ δεν χρησιμοποιεί το then που χρησιμοποιείται σε άλλες γλώσσες

- (ΕΆΝ ισχύει (δηλ. είναι αληθης)…….
  ΤΟΤΕ……..)

Έλεγχος των συνθηκών σε μια IF

Μια συνθήκη μπορεί να είναι <u>σύνθετη</u> και να περιέχει

Αριθμητικές πράξεις, τελεστές σύγκρισης και λογικούς τελεστές

Εκτελούνται πρώτα

•Πράξεις

•Τελεστές σύγκρισης

•Λογικοί τελεστές

Από αριστερά προς τα δεξιά

Λογικοί τελεστές

Υπάρχουν 3 λογικοί τελεστές

• <u>το λογικό «και» AND</u>

Συμβολίζεται && if (x>2 && x<5) k=100;

Εκτελείται όταν ισχύουν ΚΑΙ οι δυο συνθήκες

ΠΡΟΣΟΧΗ if (x>2 && <5) είναι λάθος k=100;

•<u>Το λογικό «όχι» NOT</u>

Συμβολίζεται ! If (! (x<3)) a=b;

Εκτελείται όταν δεν ισχύει η συνθήκη

•<u>Το λογικό «ή» OR</u>

Συμβολίζεται || if (x<3 || y<0) a=1;

•Εκτελείται όταν ισχύει τουλάχιστον μια συνθήκη

Συνδυασμοί λογικών τελεστών

Σε πολύπλοκους συνδυασμούς προτεραιότητα έχουν τα

NOT AND OR XR (Να Χρησιμοποιείτε παρενθέσεις!!!!)

Τελεστές ίδιας προτεραιότητας

Σε τελεστές ίδιας προτεραιότητας

Η εκτέλεση γίνεται από αριστερά προς τα δεξιά

```
If (x<10  &&  y> a*b ||  x = = 1)  d=0;
```

Να υπολογιστούν οι ρίζες του τριωνύμου… $\alpha x^2+\beta x+\gamma=0$

```
/*ypologismos rizon trionimoy*/
#  include <iostream.h>
#  include <math.h>
#  include <stdio.h>
Void main ( )
{
float a,b,c;
float d;
float x1,x2;
Printf("Dwse tin timi tou a:\n");
scanf("%f",&a);
Printf("Dwse tin timi tou b:\n");
scanf("%f",&b);
```

Να υπολογιστούν οι ρίζες του τριωνύμου… $\alpha x^2+\beta x+\gamma=0$

```
printf("Dwse tin timi tou c:\n");
scanf("%f",&c);
if(a= =0&&b!=0){
x1=-c/b;
printf("\nI eksiswsi  prokuptei na einai prwtou vathmou kai x=%f",x1);
}
if(a= =0&&b= =0)
printf("\nI eksiswsi isxuei mono ean c=0");
if(a!=0){
d=pow(b,2)-4*a*c;
if(d>0){
x1=(-b+sqrt(d))/2*a;
x2=(-b-sqrt(d))/2*a;
printf("I prwti riza einai i:%f\n",x1);
printf("I deuteri riza einai i:%f\n",x2);
```

Να υπολογιστούν οι ρίζες του τριωνύμου… $\alpha x^2+\beta x+\gamma=0$

```
if(d= =0){
x1=x2=-b/2*a;
printf("I eksiswsi exei mia dipli riza tin:%f",x1);
}
if(d<0)
printf("I eksiswsi exei duo migadikes rizes");

}

exit(0);
}
```

Nested if (Συνθήκη υπό συνθήκη)

(if υπό άλλη συνθήκη, υπό τον έλεγχο μιας άλλης if)

- If (συνθήκη-1)
  - If (συνθήκη-2)

    ……; **(περιοχή ελέγχου)**

  - **Η περιοχή ελέγχου εκτελείται όταν ισχύουν (αληθείς)**

  η συνθήκη-1 <u>και</u> η συνθήκη-2

  <u>Ή</u> ακόμη

  If (συνθήκη-1)

  If (συνθήκη-2)

  {…….. ;

  ……..; **(περιοχή ελέγχου)**

  …….. ;}

If-else

- If (συνθήκη)

  ……..; (1)

  Else

  ……..; (2)

  ……..; (3)

- Αν η συνθήκη αληθεύει τότε εκτελείται η εντολή (1) & η επόμενη εντολή που εκτελείται είναι η (3) εκεί μεταφέρεται ο έλεγχος του προγράμματος. Αγνοείται η (2)
- Αν η συνθήκη δεν αληθεύει τότε εκτελείται η εντολή (2) & η επόμενη εντολή που εκτελείται είναι η (3) εκεί μεταφέρεται ο έλεγχος του προγράμματος. Αγνοείται η (1)

Σύνθετη μορφή if -else

- If (συνθήκη)

```
{ ............... ;
  ................;    (1)
  ................;
 else
............... ;
  ................;    (2)
  ................;
}
..................;    (3)
```

Σε ποια if αντιστοιχεί η else

Σε σύνθετες περιπτώσεις if και else :

Η else πηγαίνει με την τελευταία if που δεν έχει δικό της else

<u>Π.χ1.</u> if (συνθηκη 1)

if (συνθηκη 2)

…….;

Else

…….; Εδώ η else πάει με την if (2)

<u>Π.χ2</u>. if (συνθηκη 1)

{ if (συνθηκη 2)

…….;

}

Else

…….; Εδώ η else πάει με την if (1), μη ορατή για την f(2)

Ο βρόχος FOR

Μεταβλητή ελέγχου της for: i

For (i=1; i<=n; i+=1)

αρχική τιμή μεταβλητής ελέγχου **τελική τιμή (συνθήκη)** **ρυθμός μεταβολής**

{ ……

…… **(περιοχή/πεδίο ελέγχου της for)**

…….. }

<u>Για όσο</u> ισχύει η συνθήκη εκτελούμε τις εντολές που ακολουθούν μέσα σε άγκιστρα

ΑΛΓΟΡΙΘΜΟΣ ΕΚΤΕΛΕΣΗΣ ΤΗΣ FOR

Βήμα 1: πρώτα παίρνει τιμή η μεταβλητή της FOR

Βήμα 2: μετά ελέγχεται η συνθήκη που ακολουθεί

-Αν η συνθήκη **δεν αληθευει**, σταματά η εκτέλεση της FOR και το πρόγραμμα συνεχίζει από την εντολή μετα το πεδίο έλεγχου της FOR

-Αν η συνθήκη **αληθεύει,**

εκτελούνται οι εντολές του πεδίου ελέγχου της FOR

μεταβάλλεται ο μετρητής

Επιστρέφουμε στο ΒΗΜΑ-2

```
Int I;

FOR (I=1;  I<=5; I=I+1)

{printf(/n/n "τυπωσε το %d", I)

}
```

Μετρητής

i=i+1;

i++; Η αύξηση κατά 1 θα γίνει αφού εκτελεστεί

++i; Η αύξηση κατά 1 θα γίνει πριν εκτελεστεί

i--; Η μείωση κατά 1 θα γίνει αφού εκτελεστεί

-- i; Η μείωση κατά 1 θα γίνει πριν εκτελεστεί

a=a+b; Η τιμή του a αυξάνει κατά b (βήμα/ρυθμός αύξησης b)

a+=b; Η τιμή του a αυξάνει κατά b (βήμα/ρυθμός αύξησης b)

a-=b; Η τιμή του a μειώνεται κατά b (βήμα/ρυθμός μείωσης b)

Τύπωσε άρτιους αριθμούς 2, N

```
Int  I, N;

For (I=2; I<=N; I=I+2)

{

Printf(/n/n %d, I)

}
```

Κάποιες εντολές for (πεδία ελέγχου) <u>δεν εκτελούνται</u> ποτέ γιατί δεν μπορούμε να πάμε από την αρχική τιμή της μεταβλητής ελέγχου στην τελική τιμή

for (I=1; I<=50; I--)

for (j=10; j>=0; j++)

FOR με πολλές μεταβλητές ΑΛΛΑ ΜΟΝΟ ΜΙΑ ΣΥΝΘΗΚΗ

Σε μια for μπορούμε να έχουμε περισσότερες μεταβλητές με αρχικές τιμές και αντίστοιχες εντολές που μεταβάλλουν τις τιμές τους

Αλλά έχουμε πάντοτε ΜΙΑ ΜΟΝΟ ΣΥΝΘΗΚΗ

```
INT I,J,K;

FOR (I=0, J=2;      I+J<=22;      I=I+1, J=J+1)

{K=I+J;

printf(/n/n K=%d, K);}
```

Πολλαπλά for

•Μπορούμε να έχουμε εντολές for μέσα σε for αρκεί οι εξωτερικές να επικαλύπτουν **πλήρως** τις εσωτερικές

```
•For  (I=0; I<=n; I++)
{……….;
 ……..;
        for (j=0;  j<=n; j++)
        {……;
         ……..;     Εσωτερική for
        ……..;}
……..;
……..;}
```

Εντολές τύπου WHILE

Απλή while

WHILE (συνθήκη) ……..;

Σύνθετη while

WHILE (συνθήκη)

{……..;

……..; Πεδίο ελέγχου της while

……..;} ΠΡΟΣΟΧΗ χωρις ;

Εντολή WHILE

WHILE (συνθήκη)

{……..;

……..; Πεδίο ελέγχου της while

……..;}

Στην περίπτωση αυτή εκτελούνται συνεχώς οι εντολές που ανήκουν στο πεδίο ελέγχου της while, όσο ισχύει η συνθήκη

ΑΛΛΑ η συνθήκη μεταβάλλεται και αυτή μέσα στο πεδίο ελέγχου

int I=0; int B=10;

While (I<b)

{ I=I+3; printf I}

ΑΛΓΟΡΙΘΜΟΣ ΕΚΤΕΛΕΣΗΣ WHILE

Βήμα –1 ελέγχεται η συνθήκη

Αν η συνθήκη **δεν αληθεύει** , τότε η εκτέλεση συνεχίζεται από την πρώτη εντολή μετά το πεδίο ελέγχου της WHILE

Αν η συνθήκη **αληθεύει,** εκτελούνται οι εντολές στο πεδίο ελέγχου της While

Βήμα-2: Πάμε στο Βήμα-1

Στις for η έξοδος επιτυγχάνεται όταν τελειώσουν οι επαναλήψεις, ενώ στις while μπορούμε να βγούμε μόλις πάψει να ισχύει η συνθήκη

Nested while

Μπορούμε να έχουμε μια while μέσα σε άλλη, αρκεί η μια να καλύπτει πλήρως την άλλη

While (συνθήκη-1)

{……..;

……..;

While (συνθήκη-2)

{……..;

……..;} εσωτερική while

……..;

……;}

Εντολές DO WHILE *Βρόχος do While*

Η συνθήκη εξετάζεται στο τέλος του πεδίου ελέγχου του βρόγχου. ΟΠΟΤΕ, οι εντολές του πεδίου εκτελούνται τουλάχιστον ΜΙΑ φορά.

Do

{……;

……; Πεδίο έλεγχου της do while

……;}

While (συνθήκη); ΠΡΟΣΟΧΗ **χρειάζεται ;**

Αλγόριθμος εκτέλεσης της do while

ΒΗΜΑ-1: εκτελούνται οι εντολές του πεδίου do while

ΒΗΜΑ-2: εξετάζεται η συνθήκη

αν ΔΕΝ ΑΛΗΘΕΥΕΙ σταματά η εκτέλεση της DO WHILE και συνεχίζουμε με την εντολή μετά την WHILE (συνθήκη);

αν ΑΛΗΘΕΥΕΙ, πηγαίνουμε στο ΒΗΜΑ-1

παράδειγμα

```
DO
{printf( \n "diavase to  x");
Scanf (%d, &x);
S=s+x;
I=I+1;}
While  (I<=n);
```

Θα διαβάσει τουλάχιστον ένα χ

Ασκήσεις if & for

Έστω 7 διάμετροι δένδρων: $x_{1...7}$=10,14,18,5,12,16,5. Να βρεθεί ο Μ.Ο=Σx/7

```
/* Υπολογισμός Μ.Ο συστάδας δένδρων*/
 #  include <stdio.h>
#  include <math.h>
Void main ( )
{ Int i, s, x; //Δίνω τις μεταβλητές
float mo; // Μέσος όρος
s=0; //Μηδενίζω αρχικά το άθροισμα
for (i=1; i<=7; i + =1)
{ Printf ("\n\n Διάβασε μια μέτρηση");
Scanf ("%d", &x);
s= s + x; //Προσθέτω τη νέα μέτρηση στο άθροισμα
}
mo= s/7; //Υπολογισμός Μ.Ο
Printf("\n\n Ο Μ.Ο είναι : %f  cm", mo);
}
Exit(0) ;
```

Να Υπολογιστεί ο V των κορμοτεμαχίων & εκτυπωθούν τα D,L όταν V=1 m^3. D=20,30,…,80 cm & L=3,4,…,10 m. Ισχύει $V=\pi*d^2*L/4$

```
/* Υπολογισμός Όγκου κορμοτεμαχίων*/
 #  include <iostream.h>
#  include <stdio.h>
#  include <math.h>
Void main ( )
{ Int d, L; //Δίνω τις μεταβλητές για διάμετρο και μήκος
float V,pi; // Όγκος και π
pi=3.14;
for (d=20; d<=80; d+=10)
            {           for (L=3; L<=10; L+=1)
                        {V=(pi/4)*pow(d/100,2)*L;
                                    if (V= =1) ή καλύτερα   if (V<=1.1)AND(V>=0.9)
                                    {Printf ("\n\n για διάμετρο %d",d);
                                    Printf ("\n\n για μήκος %d",L);
                                    Printf ("\n\n ο όγκος είναι %f",V);}
                        }
            }
}
Exit(0);
```

Αθροισμα Σ = 1+2+3+4+5+……+n

```
/* Αθροισμα Σ = 1+2+3+4+5+……+n */
#include <iostream.h>
#include <stdio.h>
Void main ( )
{
Int i, s, n;
s=0; // midenismos athrismatos
Printf ("\n\n dose teliko n…");
Scanf ("%d", &n);
For (i=1; i<=n; i=i+1)
{
s=s+i;
}

Printf ("\n\n to athroisma einai %d",s);
}
Exit(0);
```

Αθροισμα Σ = 1+3+5+……+n */

```
/* Αθροισμα Σ = 1+3+5+……+n */
#include <iostream.h>
#include <stdio.h>
Void main ( )
{
Int i, s, n;
s=0; // midenismos athrismatos
Printf ("\n\n dose teliko n…");
Scanf ("%d", &n);
For (i=1; i<=n; i=i+2)
{
s=s+i;
}

Printf ("\n\n to athroisma einai %d",s);
}
Exit(0);
```

Αθροισμα Σ = 2+4+6+8+……+n

```
/* Αθροισμα Σ = 2+4+6+8+……+n */
#include <iostream.h>
#include <stdio.h>
Void main ( )
{
Int i, s, n;
s=0; // midenismos athrismatos
Printf (“\n\n dose teliko n…”);
Scanf (“%d”, &n);
For (i=2; i<=n; i=i+2)
{
s=s+i;
}

Printf (“\n\n to athroisma einai %d”,s);
}
Exit(0);
```

Αθροισμα $\Sigma = 1^2+3^2+5^2+\ldots\ldots\ldots+n^2$

```
/* Αθροισμα Σ = 1²+3²+5²+……+n² */
#include <iostream.h>
#include <stdio.h>
#include <math.h>
Void main ( )
{
Int i, s, n;
s=0; // midenismos athrismatos
Printf ("\n\n dose teliko n…");
Scanf ("%d", &n);
For (i=1; i<=n; i=i+2)
{
s=s+pow(i,2);
}
Printf ("\n\n to athroisma einai %d",s);
}
Exit(0);
```

Αθροισμα Σ = $2^3+4^3+6^3+8^3+\ldots\ldots\ldots+n^3$

```
/* Αθροισμα Σ = 2³+4³+6³+8³+……+n³ */
#include <iostream.h>
#include <stdio.h>
#include <math.h>
Void main ( )
{
Int i, s, n;
s=0; // midenismos athrismatos
Printf (“\n\n dose teliko n…”);
Scanf (“%d”, &n);
For (i=2; i<=n; i=i+2)
{
s=s+pow( i , 3);
}
Printf (“\n\n to athroisma einai %d”,s);
}
Exit(0);
```

Να Υπολογιστεί ο συνολικός όγκος V των κορμοτεμαχίων & εκτυπωθούν τα D,L όταν V=1 m^3. D=20,30,…,80 cm & L=3,4,…,10 m. Ισχύει $V=\pi*d^2*L/4$

```
{ Int d, L; //Δίνω τις μεταβλητές για διάμετρο και μήκος
float V,pi; // Όγκος και π
pi=3.14;
for (d=20; d<80; d+=10)
{
for (L=3; L<=10; L+=1)
{
V=(pi/4)*pow(d/100,2)*L;
if (V=1) ή καλύτερα   if (V<=1.1)AND(V>=0.9)
    {
Printf ("\n\n για διάμετρο %d",d);
Printf ("\n\n για μήκος %d",L);
Printf ("\n\n Όγκος= %f",V);
    }
}
}
```

Επαναληπτικές ασκήσεις

Γλώσσα προγραμματισμού C++

Επαναληπτικές Ασκήσεις

Εκτύπωση άρτιων αριθμών 2,…….,n. Να τυπώνεται:

```
#  include <stdio.h>
Void main ( )
{ Int num, n, j;
j=0; //Μηδενίζω το μετρητή
Printf (“\n\n dwse to euros emfanisis artiwn:”);
Scanf (“%d”, &n);//διαβάζει το n
        for (num=2; num<=n; num+=2)
        { j++;
        Printf(“\n\n %d os ARTIOS = %d”,  j, num);
        }
}
Exit(0) ;
```

1os ARTIOS = 2

2os ARTIOS = 4

3os ARTIOS = 6

……..

Να Υπολογιστεί ο V των κορμοτεμαχίων & εκτυπωθούν τα D,L όταν V=1 m^3. D=20,30,…,80 cm & L=3,4,…,10 m. Ισχύει $V=\pi*d^2*L/4$

```
/* Υπολογισμός Όγκου κορμοτεμαχίων*/
 #  include <iostream.h>
#  include <stdio.h>
#  include <math.h>
Void main ( )
{ Int d, L; //Δίνω τις μεταβλητές για διάμετρο και μήκος
float V,pi; // Όγκος και π
pi=3.14;
for (d=20; d<=80; d+=10)
            {           for (L=3; L<=10; L+=1)
                        {V=(pi/4)*pow(d/100,2)*L;
                                    if (V= =1) ή καλύτερα   if (V<=1.1)AND(V>=0.9)
                                    {Printf ("\n\n για διάμετρο %d",d);
                                    Printf ("\n\n για μήκος %d",L);
                                    Printf ("\n\n ο όγκος είναι %f",V);}
                        }
            }
}
Exit(0);
```

Επαναληπτικές Ασκήσεις

Για δάνεια: 100.000€, 150.000€, 250.000€, Επιτόκια δανεισμού ρ=2,2 1,8 1,6 και 1,5 και για αριθμό δόσεων 18, 24, 36, 60 και 120, να υπολογίσετε το ύψος της ετήσιας δόσης R από τον τύπο: R= k . 0,0p . 1,0p^n / 1,0p^n - 1

```
/* Askisi Ypologismou miniais dosis daneiou*/
 #  include <stdio.h>
#  include <math.h>
Void main ( )
{ Int n; //, διάρκεια δανείου σε μήνες
long int k; // Κεφάλαιο
float p,r,a,r1,b; //επιτόκιο
Printf ("\n\n dwse dedomena daneiou");
Printf ("\n\n kefalaio:");
Scanf ("%1d", &k);
Printf ("\n Epitokio:");//
Scanf ("%f", &p);
Printf ("\n xroniki diarkeia se mines");
Scanf ("%d", &n);
b=p/1200; //Το επιτόκιο διαιρείται με 1200 γιατί το δάνειο εξοφλείται σε μηνιαίες δόσεις
a= pow( (1+b) , n);
```

Επαναληπτικές Ασκήσεις

```
/* Askisi Ypologismou miniais dosis daneiou*/
 #  include <stdio.h>
#  include <math.h>
Void main ( )
{ Int p, n; //επιτόκιο, διάρκεια δανείου σε μήνες
long int k; // Κεφάλαιο
float r,a,r1,b;
Printf ("\n\n dwse dedomena daneiou");
Printf ("\n\n kefalaio:");
Scanf ("%1d", &k);
Printf ("\n Epitokio:");// πχ 18 και όχι 0.18
Scanf ("%d", &p);
Printf ("\n xroniki diarkeia se mines");
Scanf ("%d", &n);
b=p/1200; //Το επιτόκιο διαιρείται με 1200 γιατί το δάνειο εξοφλείται σε μηνιαίες δόσεις
a= pow( (1+b) , n);
r1= (k*b*a);
r = r1/(a-1.0);
Printf ("\n\n miniaia dosi = %10.2f \n" , r);
Exit() }
```