



Πολυτεχνική Σχολή
Τμήμα Μηχανικών Η/Υ & Πληροφορικής
Προπτυχιακό Πρόγραμμα Σπουδών

1.

Εξοικείωση με τη σύνταξη κώδικα

Αυτό το κεφάλαιο καλύπτει εισαγωγικές έννοιες κι ενέργειες κατανόησης και εξοικείωσης με τη χρήση του μεταγλωττιστή gcc, τη σύνταξη κώδικα, την απασφαλμάτωση και μια βασική γνωριμία με τύπους δεδομένων, τελεστές και παραστάσεις.

1.1. Εισαγωγική εξάσκηση

Χρησιμοποιήστε ένα editor (π.χ. pico, nano, vi) και γράψτε το παρακάτω μικρό πρόγραμμα το οποίο θα αποθηκεύετε στο αρχείο example.c.

```
#include <stdio.h>
/*
* example program
*/
int main()
{
      int a;
      int b;
      a = 2;
      b = 1;
      printf("%d\n",a);
      printf("%d\n",b);
      scanf("%d",&a);
      scanf("%d",&b);
      printf("%d\n",a);
      printf("%d\n",b);
}
```

- Μεταγλωττίστε το πρόγραμμά σας χρησιμοποιώντας την εντολή: gcc example.c και τρέξτε το πρόγραμμα σας:

```
./a.out
```

- Μεταγλωττίστε το πρόγραμμά σας χρησιμοποιώντας την εντολή: gcc –o example example.c και τρέξτε το πρόγραμμα σας:

```
./example
```

1.2. Συχνά Λάθη

Κάντε με τη σειρά τις παρακάτω αλλαγές που λέει το πρόγραμμα και δείτε τα μηνύματα λάθους που βγάζει ο μεταγλωττιστής. Αφού δοκιμάσετε κάποιο λάθος επαναφέρετε το όπως ήταν αρχικά για να δοκιμάσετε το επόμενο λάθος στην λίστα. Εντοπίστε σε ποιο λάθος ποιος κανόνας της γλώσσας παραβιάζεται και σε ποιες περιπτώσεις βγάζει λάθος ο μεταγλωττιστής και σε ποιες προειδοποίηση.

- Διαγράψτε την εντολή του προεπεξεργαστή #include <stdio.h>
- Αλλάξτε την μεταβλητή a σε A στην εντολή a=2
- Σβήστε την δήλωση int a
- Αλλάξτε την εντολή main σε new_main
- Σβήστε το }
- Μετακινήστε την δήλωση int a, μετά την εντολή a=2;
- Σβήστε το “;” στο τέλος της δήλωσης int a;

1.3. Διόρθωση Λαθών

Διορθώστε τα λάθη στο παρακάτω πρόγραμμα ώστε να μεταγλωττίζεται χωρίς λάθη και προειδοποιήσεις και να πραγματοποιεί σωστά τις αριθμητικές πράξεις ανάμεσα στα στοιχεία που υποστηρίζει ότι πραγματοποιεί.

```
#include <stdio.h> int Main()
{
    int x,y /* dilosi metablitwn */
    /* eisodos twn 2 akeraiwn apo to std input: */ printf("Doste x: ");
    scanf("%d",x)
    printf("Doste y: ");
    scanf("%d",&y);
    /* ektelesi kai typwsi apotelesmatwn sto std output */
    printf("%d + %d = %d\n",x,y,x+y); printf("%d - %d = %d\n",x,y,x-y); printf("%d * %d
    =  %d\n",x,y,x*y);  printf("%d  /  %d  =  %d\n",x,y,x/y);  pintf("%d  mod  %d
    = %d\n",x,y,x%y); return 0;
}
```

1.4. Απλές εφαρμογές και προγράμματα εξοικείωσης

1. Να σχεδιάσετε αλγόριθμο σε ψευδογλώσσα και να κατασκευάσετε τα βήματα του σε ANSI C, που:
    - Θα διαβάζει από το πληκτρολόγιο τις ακαθάριστες αποδοχές ενός υπαλλήλου,
    - Θα υπολογίζει και θα τυπώνει σε ξεχωριστές γραμμές στην οθόνη, τις ασφαλιστικές εισφορές, τον αναλογούντα φόρο και τις καθαρές μηνιαίες αποδοχές του υπαλλήλου.

    Δίνεται ότι το ποσοστό των ασφαλιστικών εισφορών είναι το 30% των ακαθάριστων αποδοχών και ο ανάλογος φόρος το 10% του ποσού (ακαθάριστες αποδοχές μείον ασφαλιστικές εισφορές).

    Υπόδειγμα: Οι τύποι δεδομένων των μεταβλητών και σταθερών που θα χρησιμοποιήσετε αφήνονται στην κρίση σας, αλλά πρέπει να αιτιολογήσετε τις αποφάσεις σας για την επιλογή τους και να επιβεβαιώσετε την ορθότητα και την ακρίβεια τους.

```
ΑΛΓΟΡΙΘΜΟΣ μισθοδοσία

        ΠΡΑΓΜΑΤΙΚΕΣ: ακαθάριστες, εισφορές, φόρος, καθαρέςαποδοχες

        ΓΡΑΨΕ 'Δώσε ακαθάριστες αποδοχές:'
        ΔΙΑΒΑΣΕ ακαθάριστες
        εισφορές = e * ακαθάριστες
        φόρος = (ακαθάριστες - εισφορές) * f
        μισθός = ακαθάριστες - εισφορές - φόρος
        ΓΡΑΨΕ 'Κρατήσεις:', εισφορές
        ΓΡΑΨΕ 'Φόρος', φόρος
        ΓΡΑΨΕ 'Καθαρές αποδοχές:', μισθός
ΤΕΛΟΣ μισθοδοσία
```

```
#include <stdio.h>

double akatharistes, eisfores, foros, katharesapodoxes;

int main()
```

```
{
        printf("Dose tis akatharistes apodoxes: ");
        scanf("%lf", &akatharistes);
        eisfores=akatharistes*0.3;
        foros=(akatharistes-eisfores)*0.1;
        katharesapodoxes=akatharistes-eisfores-foros;
        printf("Krathseis: %lf\n", eisfores);
        printf("Foros: %lf\n", foros);
        printf("Kathares Apodoxes: %lf\n", katharesapodoxes);
        return 0;
}
```

2. Ένα κατάστημα ηλεκτρικών ειδών πουλάει τα είδη του με δύο τρόπους:
   - Μετρητοίς, οπότε γίνεται έκπτωση 10% επί της αρχικής αξίας.
   - Σε 12 μηνιαίες δόσεις, οπότε η τιμή προσαυξάνεται κατά 15% και απαιτείται προκαταβολή του 10% της αρχικής αξίας.

   Να σχεδιάσετε αλγόριθμο σε ψευδογλώσσα και να κατασκευάσετε τα βήματα του σε ANSI C, που θα:
   - δέχεται την αρχική αξία μιας ηλεκτρικής συσκευής,
   - υπολογίζει την τιμή μετρητοίς της συσκευής,
   - υπολογίζει την τιμή της συσκευής με δόσεις,
   - υπολογίζει το ποσό προκαταβολής στην περίπτωση των δόσεων,
   - υπολογίζει το ποσό των μηνιαίων δόσεων,
   - δημιουργεί εκτύπωση στην οθόνη για κάθε έναν από τους παραπάνω υπολογισμούς.

   Υπόδειγμα: Οι τύποι δεδομένων των μεταβλητών και σταθερών που θα χρησιμοποιήσετε αφήνονται στην κρίση σας, αλλά πρέπει να αιτιολογήσετε τις αποφάσεις σας για την επιλογή τους και να επιβεβαιώσετε την ορθότητα και την ακρίβεια τους.

```
ΑΛΓΟΡΙΘΜΟΣ Δόσεις
        ΠΡΑΓΜΑΤΙΚΕΣ: ΑΑ, ΤΜ, ΤΔ, ΠΡΟ, ΜΔ
        ΓΡΑΨΕ 'Δώσε την αρχική αξία:'
        ΔΙΑΒΑΣΕ ΑΑ
        ΤΜ = ΑΑ – ΑΑ*10/100
        ΤΔ = ΑΑ + ΑΑ* 15/100
        ΠΡΟ = ΑΑ * 10/100
        ΜΔ = (ΤΔ – ΠΡΟ) / 12
        ΓΡΑΨΕ 'Αρχική αξία= ', ΑΑ, '€'
        ΓΡΑΨΕ 'Τιμή μετρητοίς= ', ΤΜ, '€'
        ΓΡΑΨΕ 'Τιμή δόσεων= ', ΤΔ, '€'
        ΓΡΑΨΕ 'Προκαταβολή= ', ΠΡΟ, '€'
        ΓΡΑΨΕ 'Μηνιαία δόση ', ΜΔ, '€'
ΤΕΛΟΣ Δόσεις
```

```
#include <stdio.h>

double aa, tm, td, pro, md;

int main()
{
      printf("Dose tin arxiki axia: ");
      scanf("%lf", &aa);
      tm=aa-aa*10/100;
      td=aa+aa*15/100;
      pro=aa*10/100;
      md=(td-pro)/12;
      printf("Arxiki axia: %lf €\n", aa);
      printf("Timi Metrhtois: %lf €\n", tm);
      printf("Timi doseis: %lf €\n", td);
```

```
        printf("Prokataboli: %lf €\n", pro);
        printf("Miniaia dosi: %lf €\n", md);
        return 0;
}
```

3. Να γραφεί ένα πρόγραμμα το οποίο να εκχωρεί δύο θετικές πραγματικές τιμές σε δύο πραγματικές μεταβλητές και να εμφανίζει το ακέραιο μέρος του πηλίκου τους και το υπόλοιπο της διαίρεσής τους με τη μορφή δεκαδικού αριθμού. Για παράδειγμα, αν εκχωρηθούν οι τιμές 7.2 και 5.4, το πρόγραμμα να εμφανίζει 1 και 1.8, αφού 7.2 = (1×5.4)+1.8.

```
#include <stdio.h>
int main(void)
{
        int i;
        float j, k;

        j = 7.2;
        k = 5.4;
        i = j/k;
        printf("%d %f\n", i, j-(i*k));
        return 0;
}
```

4. Να κατασκευάσετε πρόγραμμα σε ANSI C, το οποιο με δεδομένες τις τιμές των μεταβλητών x=3, y=3, z=1, w=15, a=14, b=3, να υπολογιστεί το αποτέλεσμα των παρακάτω έξι (6) παραστάσεων και να δοθεί η σειρά των βημάτων που ακολουθεί ο μεταγλωττιστής για τον υπολογισμό.
    i. res = x+y+z/x;
    ii. res = z+w*x/y/z;
    iii. res = w/x*++z+x/y;
    iv. res = ++b, --a; v. res = (--b, ++a);
    v. res = (a>b)?b:a;

```
#include <stdio.h>

int main()
{
        int x,y,z,w,a,b,res;

        x=3; y=3; z=1; w=15; a=14; b=3;

        res = y+z/x;
        printf("i.res: %d\n", res); /*res=3, καθώς πρώτα γίνεται η πράξη z/x=1/3=0
        (ακέραια διαίρεση) και μετά res=y (res=3) */

        res = w*x/y/z;
        printf("ii.res: %d\n", res); /* res=15, καθώς πρώτα γίνονται οι πράξεις *
        και / είναι ίδιας προτεραιότητας, εκτελούνται
        όμως από αριστερά προς τα δεξιά. Πρώτα γίνεται η πράξη w*x=45 μετά
        45/y=45/3=15 και τέλος 15/z=15/1=15 */

        res = w/x*++z+x/y;
        printf("iii.res: %d\n", res); /* res=4 καθώς πρώτα το z αυξάνει κατά 1
        (z=2) ,μετά γίνεται η πράξη w/x=15/3=5 , με τα 5*2=10 με τα z/y=3/3=1 και
        μετά 10+1=11 */

        res = ++b, --a;
        printf("iv.res: %d\n", res); /* res=4 καθώς πρώτα το b αυξάνει κατά 1
        (γίνεται 4) μετά το α μειώνεται κατά 1 (γίνεται 13) και επειδή το = έχει
```

```
        μεγαλύτερη προτεραιότητα από το , γίνεται το res 4 και μετά απλώς
        εμφανίζεται η τιμή α */

        res = (--b, ++a);
        printf("vi.res: %d\n", res); /* res = 14 καθώς πρώτα μειώνεται το b κατά 1
        (γίνεται 3) μετά το α αυξάνεται κατά 1 (γίνεται 14) μετά εκτελείται το ,
        (λόγω παρένθεσης) και όλη η παράσταση αποτιμάται στο δεύτερο όρισμα (14)
        και μετά ανατίθεται η τιμή στο res (14) */

        res = (a>b)?b:a; /* res= 3 καθώς το α είναι μεγαλύτερο από το β */
        printf("i.res: %d\n", res);

        return 0;
    }
```

5. Να γραφεί πρόγραμμα σε C το οποίο διαβάζει από το πληκτρολόγιο έναν πραγματικό αριθμό. Στην συνέχεια, θεωρώντας ότι ο αριθμός αυτός αναπαριστά θερμοκρασία σε βαθμούς Φαρενάιτ, να γίνει μετατροπή σε κλίμακα Κελσίου και το αποτέλεσμα να τυπωθεί.
   Υπόδειγμα: ο τύπος μετατροπής από Φαρενάιτ σε Κελσίου είναι C = (5/9) * (F - 32)

```
#include <stdio.h>

int main(void)
{
      float f,c;
      printf("Doste thermokrasia se far.: ");
      scanf("%f",&f);
      c = (5.0/9.0)*(f-32.0);
      printf("Thermokrasia se celcius: %.4f\n",c);
      return 0;
}

/* Αν αντί για (5.0/9.0) στην γραμμή 7 γράφαμε (5/9) το αποτέλεσμα θα ήταν πάντα
μηδενικό γιατί το 5/9 είναι διαίρεση ακεραίων. */
```

6. Η εταιρεία κινητής τηλεφωνίας Tmob χρεώνει τις υπηρεσίες τηλεφωνίας που παρέχει στους συνδρομητές της ως εξής:
   - Πάγια χρέωση: 12 €
   - Συνομιλία: 0.02 € / δευτερόλεπτο(χωρίς ελάχιστη διάρκεια χρέωσης)
   - SMS: 0.14 €/μήνυμα

   Σημειώνεται ότι στις παραπάνω τιμές θα προστίθεται ΦΠΑ23%. Να αναπτύξετε πρόγραμμα που:
   i. Θα διαβάζει από το πληκτρολόγιο τον τετραψήφιο κωδικό πελάτη, το χρόνο ομιλίας του σε δευτερόλεπτα και τον αριθμό των μηνυμάτων που έστειλε.
   ii. Θα υπολογίζει το ποσό του λογαριασμού σύμφωνα με την εκφώνηση.

```
#include <stdio.h>

int main()
{
      int code, sec, sms, pagio;
      float sum, fpa;
      pagio = 12;
      printf("Δώστε τον τετραψήφιο κωδικό του πελάτη: ");
      scanf("%d", &code);
      printf("Δώστε το χρόνο ομιλίας του πελάτη σε δευτερόλεπτα: ");
```

```
	scanf("%d", &sec);
	printf("Δώστε τα SMS του πελάτη: ");
	scanf("%d", &sms);
	sum = pagio + 0.02 * sec + 0.14 * sms; fpa = 23 * sum / 100;
	sum = sum + fpa;
	printf("Tmob account\n \b-------------------------------------------------
		----------------------\n \bCustomer code\t time\t SMS\t FPA\t |
		\tTOTAL\n");
	printf("%d\t\t %d\t %d\t %.2f\t | \t%.2f\n", code, sec, sms, fpa, sum);
	printf(" \n");
	return 0;
}
```

7. Θα εμφανίζει στην οθόνη τα προηγούμενα στοιχεία, μαζί με το συνολικό ποσό. Η έξοδος του προγράμματος στην οθόνη θα είναι μορφοποιημένη σύμφωνα με το ακόλουθο υπόδειγμα:

Tmob account

Customer code	time	SMS	FPA		TOTAL
4881	364	125	8.46		45.25

```
#include <stdio.h>

int main()
{
	int code, sec, sms, pagio;
	float sum, fpa;
	pagio = 12;
	printf("Δώστε τον τετραψήφιο κωδικό του πελάτη: ");
	scanf("%d", &code);
	printf("Δώστε το χρόνο ομιλίας του πελάτη σε δευτερόλεπτα: ");
	scanf("%d", &sec);
	printf("Δώστε τα SMS του πελάτη: ");
	scanf("%d", &sms);
	sum = pagio + 0.02 * sec + 0.14 * sms; fpa = 23 * sum / 100;
	sum = sum + fpa;
	printf("Tmob account\n \b-------------------------------------------------
		----------------------\n \bCustomer code\t time\t SMS\t FPA\t |
		\tTOTAL\n");
	printf("%d\t\t %d\t %d\t %.2f\t | \t%.2f\n", code, sec, sms, fpa, sum);
	printf(" \n");
	return 0;
}
```

8. Να γραφεί ένα πρόγραμμα το οποίο να διαβάζει έναν διψήφιο ακέραιο και να δημιουργεί έναν τετραψήφιο ακέραιο διπλασιάζοντας την εμφάνιση του κάθε ψηφίου του. Για παράδειγμα, αν ο χρήστης εισάγει 24, το πρόγραμμα να εμφανίζει 2244.

```
#include <stdio.h>

int main(void)
{
	int num, div, rem;
	printf("Enter a number: ");
	scanf("%d", &num); // Ανάγνωση αριθμού
	div = num/10;
	rem = num%10;
	printf("%d\n", div*1000+div*100+rem*10+rem);
	return 0;
}
```

2.

Βασικές παραστάσεις-Είσοδος/ Έξοδος

2.1. Εργαστηριακές Ασκήσεις

1. Γράψτε πρόγραμμα σε C που διαβάζει δύο πραγματικούς αριθμούς από το πληκτρολόγιο και εμφανίζει το άθροισμα, τη διαφορά, το γινόμενο και το πηλίκο τους.

   Επεκτείνετε το πρόγραμμα ώστε οι τιμές να εκτυπώνονται με δύο δεκαδικά ψηφία.

   Τροποποιήστε το πρόγραμμα ώστε να χειρίζεται πραγματικούς διπλής ακρίβειας (double).

2. Να γραφεί ένα πρόγραμμα το οποίο να εκχωρεί δύο θετικές πραγματικές τιμές σε δύο πραγματικές μεταβλητές και να εμφανίζει το ακέραιο μέρος του πηλίκου τους και το υπόλοιπο της διαίρεσής τους με τη μορφή δεκαδικού αριθμού. Για παράδειγμα, αν εκχωρηθούν οι τιμές 7.2 και 5.4, το πρόγραμμα να εμφανίζει 1 και 1.8, αφού 7.2 =(1×5.4)+1.8.

3. Να γραφεί ένα πρόγραμμα το οποίο να διαβάζει δύο θετικούς τριψήφιους ακέραιους, να αντιμεταθέτει τις εκατοντάδες του πρώτου με τις μονάδες του δεύτερου και να εμφανίζει τους νέους αριθμούς. Για παράδειγμα, αν ο χρήστης εισάγει 123 και 456 το πρόγραμμα να εμφανίζει 623 και 451. Να χρησιμοποιήσετε μέχρι τρεις μεταβλητές.

4. Να γραφεί πρόγραμμα C το οποίο:

   α. διαβάζει από το πληκτρολόγιο έναν double d

   β αποθηκεύει τον d σε έναν float f με χρήση typecasting (float) γ. τυπώνει τους d και f με ακρίβεια 10 δεκαδικών ψηφίων.

Εκτελέστε το παραπάνω πρόγραμμα δίνοντας για d την τιμή 1.123. Μετά, εκτελέστε το παραπάνω πρόγραμμα δίνοντας για d την τιμή 1.123456789123. Τί παρατηρείτε;

5. Να γραφεί πρόγραμμα C το οποίο να διαβάζει έναν τριψήφιο ακέραιο και να υπολογίζει το άθροισμα των ψηφίων του.

6. Να γραφεί πρόγραμμα σε C που διαβάζει μία γωνία σε μοίρες (degrees) και τη μετατρέπει σε ακτίνια (radians). Η μετατροπή δίνεται από τον τύπο:

$$\text{radians} = \text{degrees} \cdot \frac{\pi}{180}$$

όπου $\pi \approx 3.141592653589793$

7. Να γραφεί πρόγραμμα σε C το οποίο διαβάζει έναν ακέραιο αριθμό nκαι υπολογίζει τη δύναμη 2^n. Για τις ανάγκες της άσκησης ο υπολογισμός να γίνει μόνο με διαδοχικούς πολλαπλασιασμούς. Θεωρήστε ότι $n = 3$. Το πρόγραμμα να εμφανίζει το αποτέλεσμα της δύναμης.

8. Να γραφεί πρόγραμμα σε C το οποίο διαβάζει δύο θετικούς πραγματικούς αριθμούς aκαι b (θεωρούμε ότι το $b \neq 0$) και υπολογίζει το υπόλοιπο της διαίρεσης του aμε το b. Το υπόλοιπο να υπολογιστεί χωρίς χρήση του τελεστή %, καθώς αυτός εφαρμόζεται μόνο σε ακέραιους αριθμούς.

9. Να γραφεί πρόγραμμα σε C το οποίο διαβάζει έναν ακέραιο αριθμό που εκφράζει συνολικά δευτερόλεπτα και τα μετατρέπει σε ώρες σε δεκαδική μορφή. Το πρόγραμμα να εμφανίζει τις ώρες με ακρίβεια 3 δεκαδικών ψηφίων.

10. Να γραφεί πρόγραμμα σε C το οποίο διαβάζει έναν θετικό τριψήφιο ακέραιο αριθμό και κατασκευάζει έναν νέο αριθμό όπου τα ψηφία του αρχικού αριθμού εμφανίζονται με αντίστροφη σειρά και αυξημένα κατά 1.