1ο Φροντιστήριο Δομημένου Προγραμματισμού

Παράδειγμα 1: Πρόγραμμα ελέγχου αριθμού σε C

Σκοπός του προγράμματος

Το πρόγραμμα ζητά από τον χρήστη να εισάγει έναν ακέραιο αριθμό και στη συνέχεια:

1. Ελέγχει αν ο αριθμός είναι **θετικός**, **αρνητικός** ή **μηδέν**.
2. Αν ο αριθμός **δεν είναι μηδέν**, ελέγχει επιπλέον αν είναι **άρτιος** ή **περιττός**.

Με αυτό τον τρόπο, ο χρήστης λαμβάνει πλήρη πληροφόρηση για τη φύση του αριθμού που εισήγαγε.

Κώδικας προγράμματος

```c
#include <stdio.h>

int main() {
    int num;

    printf("Δώσε έναν αριθμό: ");
    scanf("%d", &num);

    if (num > 0) {
        printf("Ο αριθμός είναι θετικός.\n");
    } else if (num < 0) {
        printf("Ο αριθμός είναι αρνητικός.\n");
    } else {
        printf("Ο αριθμός είναι μηδέν.\n");
    }

    if (num != 0) {
        if (num % 2 == 0) {
            printf("Ο αριθμός είναι άρτιος.\n");
        } else {
            printf("Ο αριθμός είναι περιττός.\n");
        }
    }

    return 0;
}
```

Εντολή #include <stdio.h>

Η γραμμή αυτή εισάγει τη βιβλιοθήκη stdio.h, η οποία παρέχει τις συναρτήσεις **εισόδου και εξόδου** (input/output) όπως:

- printf() για εκτύπωση στην οθόνη
- scanf() για ανάγνωση τιμών από τον χρήστη

Χωρίς αυτή τη βιβλιοθήκη, το πρόγραμμα δεν θα μπορούσε να διαβάζει ή να εμφανίζει δεδομένα.

Δήλωση μεταβλητής

int num;

Η μεταβλητή num είναι τύπου int (ακέραιος αριθμός) και χρησιμοποιείται για να αποθηκεύσει τον αριθμό που θα δώσει ο χρήστης.

Εντολές εισόδου και εξόδου

printf("Δώσε έναν αριθμό: ");

scanf("%d", &num);

- Η printf() εμφανίζει ένα μήνυμα στην οθόνη ζητώντας από τον χρήστη να εισάγει έναν αριθμό.
- Η scanf() διαβάζει την τιμή που πληκτρολογεί ο χρήστης και την αποθηκεύει στη μεταβλητή num.
    - Το %d δηλώνει ότι η είσοδος είναι **ακέραιος**.
    - Το &num σημαίνει ότι η τιμή θα αποθηκευτεί στη **διεύθυνση μνήμης** της μεταβλητής num.

Δομές Ελέγχου (if, else if, else)

Η δομή if χρησιμοποιείται για να **ελέγχει συνθήκες** (δηλαδή αν κάτι είναι αληθές ή ψευδές).

Πρώτος έλεγχος: Θετικός, αρνητικός ή μηδέν

```
if (num > 0) {
    printf("Ο αριθμός είναι θετικός.\n");
} else if (num < 0) {
    printf("Ο αριθμός είναι αρνητικός.\n");
} else {
    printf("Ο αριθμός είναι μηδέν.\n");
}
```

- Αν num > 0, εκτελείται το πρώτο μπλοκ και εκτυπώνεται ότι ο αριθμός είναι θετικός.
- Αν η πρώτη συνθήκη δεν ισχύει, ελέγχεται η επόμενη (num < 0).
- Αν ούτε αυτή ισχύει, τότε εκτελείται το else, που σημαίνει ότι ο αριθμός είναι **μηδέν**.

Αυτή η δομή εξασφαλίζει ότι **μόνο μία** από τις τρεις περιπτώσεις θα εκτελεστεί.

Δεύτερος έλεγχος: Άρτιος ή περιττός

```
if (num != 0) {
    if (num % 2 == 0) {
        printf("Ο αριθμός είναι άρτιος.\n");
    } else {
        printf("Ο αριθμός είναι περιττός.\n");
    }
}
```

- Ελέγχουμε πρώτα αν num != 0, δηλαδή αν ο αριθμός **δεν είναι μηδέν**, γιατί το μηδέν δεν χρειάζεται να χαρακτηριστεί ως άρτιο ή περιττό εδώ.
- Ο τελεστής % υπολογίζει το **υπόλοιπο της διαίρεσης** του αριθμού με το 2:
    - Αν num % 2 == 0, ο αριθμός είναι **άρτιος**.
    - Αλλιώς, είναι **περιττός**.

Εντολή return 0;

Δηλώνει ότι το πρόγραμμα ολοκληρώθηκε **επιτυχώς** χωρίς σφάλματα.

Συνοψίζοντας

Το πρόγραμμα δείχνει πώς:

- Να **διαβάζουμε δεδομένα** από τον χρήστη (scanf)
- Να **εμφανίζουμε αποτελέσματα** (printf)
- Να χρησιμοποιούμε **δομές επιλογής (if, else if, else)** για να παίρνουμε αποφάσεις μέσα στο πρόγραμμα
- Να εφαρμόζουμε **αριθμητικούς ελέγχους** (π.χ. έλεγχος πρόσημου, άρτιος/περιττός)

Παράδειγμα 2 – Έλεγχος ηλικίας (αν ο χρήστης είναι ανήλικος ή ενήλικος)

Περιγραφή

Το πρόγραμμα ζητά από τον χρήστη να εισάγει την ηλικία του και στη συνέχεια ελέγχει:

- Αν είναι μικρότερος από 18 → **Ανήλικος**
- Αν είναι 18 ή μεγαλύτερος → **Ενήλικος**

Παράδειγμα εκτέλεσης

Δώσε την ηλικία σου: 16

Είσαι ανήλικος.

Δώσε την ηλικία σου: 25

Είσαι ενήλικος.

Κώδικας

```
#include <stdio.h>

int main() {
    int age;

    printf("Δώσε την ηλικία σου: ");
    scanf("%d", &age);

    if (age < 18) {
        printf("Είσαι ανήλικος.\n");
    } else {
        printf("Είσαι ενήλικος.\n");
    }

    return 0;
}
```

Ανάλυση

- Χρησιμοποιείται η scanf για να διαβαστεί η ηλικία από τον χρήστη.
- Η δομή if ελέγχει αν η τιμή είναι μικρότερη από 18.
- Αν όχι, εκτελείται το else, που σημαίνει ότι το άτομο είναι ενήλικο.
- Έτσι βλέπεις τη βασική λογική ενός **δυαδικού ελέγχου (δύο πιθανές περιπτώσεις)**.

Παράδειγμα 2 – Έλεγχος θερμοκρασίας

Περιγραφή

Το πρόγραμμα ζητά από τον χρήστη να εισάγει μια θερμοκρασία (σε βαθμούς Κελσίου) και στη συνέχεια:

- Ελέγχει αν η θερμοκρασία είναι **κάτω από το μηδέν**, **μεταξύ 0 και 30**, ή **πάνω από 30**.
- Εμφανίζει κατάλληλο μήνυμα για τον καιρό.

Παράδειγμα εκτέλεσης

Δώσε θερμοκρασία: -5

Κάνει πολύ κρύο!

Δώσε θερμοκρασία: 20

Ο καιρός είναι καλός.

Δώσε θερμοκρασία: 35

Κάνει ζέστη!

Κώδικας

```
#include <stdio.h>

int main() {
    float temp;

    printf("Δώσε θερμοκρασία (°C): ");
    scanf("%f", &temp);

    if (temp < 0) {
        printf("Κάνει πολύ κρύο!\n");
    } else if (temp <= 30) {
        printf("Ο καιρός είναι καλός.\n");
    } else {
        printf("Κάνει ζέστη!\n");
    }

    return 0;
}
```

Ανάλυση

- Εδώ χρησιμοποιείται η μεταβλητή temp τύπου float, γιατί η θερμοκρασία μπορεί να έχει δεκαδικά ψηφία.
- Με τη scanf("%f", &temp) διαβάζεται μια δεκαδική τιμή.
- Η δομή if – else if – else επιτρέπει **τρεις διαφορετικές περιπτώσεις**.
- Η συνθήκη temp <= 30 καλύπτει όλες τις θερμοκρασίες από **0 έως 30**.

Ένα πιο σύνθετο παράδειγμα ...

Περιγραφή

Να γραφεί πρόγραμμα που υπολογίζει τον τελικό βαθμό ενός φοιτητή με βάση:

- Τρεις εργασίες (E1, E2, E3)
- Μία τελική εξέταση
- Πιθανό bonus συμμετοχής

Το πρόγραμμα πρέπει να:

1. Διαβάζει τους βαθμούς (0–10).
2. Ελέγχει με if–else αν κάθε βαθμός είναι έγκυρος.
3. Υπολογίζει τον τελικό μέσο όρο με συντελεστές:
    - Εργασίες: 60% (20% η καθεμία)
    - Τελική εξέταση: 40%
4. Αν υπάρχει bonus συμμετοχής (επιλογή με switch), να προστίθεται 0.5 μονάδα (χωρίς να ξεπερνά το 10).
5. Τέλος, να εμφανίζει με if–else if–else την αξιολόγηση:
    - 0–4.99 → "Αποτυχία"
    - 5–6.99 → "Επαρκής"
    - 7–8.99 → "Πολύ καλός"
    - 9–10 → "Άριστος"

```c
#include <stdio.h>
int main() {
    float e1, e2, e3;    // Βαθμοί εργασιών
    float exam;          // Βαθμός τελικής εξέτασης
    float finalGrade;    // Τελικός βαθμός
    int bonusChoice;     // Επιλογή για bonus
    printf("Δώσε τον βαθμό της Εργασίας 1 (0-10): ");
    scanf("%f", &e1);
    if (e1 < 0 || e1 > 10) {
        printf("Μη έγκυρος βαθμός!\n");
        return 0;
    }
    printf("Δώσε τον βαθμό της Εργασίας 2 (0-10): ");
    scanf("%f", &e2);
    if (e2 < 0 || e2 > 10) {
        printf("Μη έγκυρος βαθμός!\n");
        return 0;
    }
    printf("Δώσε τον βαθμό της Εργασίας 3 (0-10): ");
    scanf("%f", &e3);
    if (e3 < 0 || e3 > 10) {
        printf("Μη έγκυρος βαθμός!\n");
        return 0;
    }
    printf("Δώσε τον βαθμό της Τελικής Εξέτασης (0-10): ");
    scanf("%f", &exam);
    if (exam < 0 || exam > 10) {
        printf("Μη έγκυρος βαθμός!\n");
        return 0;
    }
    finalGrade = (e1 * 0.2) + (e2 * 0.2) + (e3 * 0.2) + (exam * 0.4);
    printf("Έχεις bonus συμμετοχής;\n");
    printf("1. Ναι\n");
    printf("2. Όχι\n");
    printf("Επέλεξε (1 ή 2): ");
    scanf("%d", &bonusChoice);
    switch (bonusChoice) {
        case 1:
            finalGrade += 0.5;
            if (finalGrade > 10) {
                finalGrade = 10; // Ο βαθμός δεν μπορεί να ξεπερνά το 10
            }
            break;
        case 2:
            // δεν προστίθεται bonus
            break;
        default:
            printf("Μη έγκυρη επιλογή bonus. Δεν θα προστεθεί bonus.\n");
    }
    printf("\nΤελικός βαθμός: %.2f\n", finalGrade);
    if (finalGrade < 5) {
        printf("Αξιολόγηση: Αποτυχία\n");
    } else if (finalGrade < 7) {
        printf("Αξιολόγηση: Επαρκής\n");
    } else if (finalGrade < 9) {
        printf("Αξιολόγηση: Πολύ καλός\n");
    } else {
        printf("Αξιολόγηση: Άριστος\n");
    }
    return 0;
}
```

Αναλυτική Επεξήγηση του Προγράμματος (βήμα-βήμα)

1) Δήλωση μεταβλητών — τι είναι και γιατί

```
float e1, e2, e3;     // Βαθμοί εργασιών
float exam;           // Βαθμός τελικής εξέτασης
float finalGrade;     // Τελικός βαθμός
int bonusChoice;      // Επιλογή για bonus
```

- float χρησιμοποιούμε γιατί οι βαθμοί μπορεί να είναι δεκαδικοί (π.χ. 7.5).
- int bonusChoice κρατά την επιλογή του χρήστη (1 ή 2). Δεν χρειάζεται float εδώ.

2) Είσοδος βαθμών και έλεγχος εγκυρότητας

Για κάθε βαθμό κάνουμε:

1. printf για μήνυμα προς τον χρήστη.
2. scanf("%f", &e1); για ανάγνωση.
3. Άμεσο έλεγχο if (e1 < 0 || e1 > 10) — αν είναι εκτός εύρους τερματίζουμε με return 0;.

Σκοπός: αποφεύγουμε περαιτέρω λογική σε περίπτωση μη έγκυρων δεδομένων — πιο ασφαλές και απλό.

Σημεία προσοχής:

- scanf επιστρέφει πόσες τιμές διάβασε επιτυχώς. Σε πιο «σκληρά» προγράμματα ελέγχουμε και την επιστρεφόμενη τιμή της scanf.
- Αν ο χρήστης δώσει μη αριθμητικά δεδομένα, η scanf αποτυγχάνει και ίσως αφήσει το πρόγραμμα σε αδιέξοδο — για πλήρη ρομπόστ έλεγχο θα χρειαστεί ανάγνωση γραμμής (fgets) και sscanf ή strtof.

3) Υπολογισμός τελικού βαθμού (στοιχείο λογικής)

```
finalGrade = (e1 * 0.2) + (e2 * 0.2) + (e3 * 0.2) + (exam * 0.4);
```

- Κάθε εργασία έχει συντελεστή 20% → 0.2. Οι τρεις μαζί δίνουν 60%.
- Η τελική εξέταση 40% → 0.4.
- Ο τύπος υπολογίζει τον σταθμισμένο μέσο όρο.

Παράδειγμα αριθμητικού υπολογισμού:

- e1 = 8.0, e2 = 6.5, e3 = 7.0, exam = 9.0
  finalGrade = 8.0*0.2 + 6.5*0.2 + 7.0*0.2 + 9.0*0.4
  = 1.6 + 1.3 + 1.4 + 3.6 = 7.9

Σημείωση: Οι δεκαδικοί υπολογισμοί με float έχουν μικρή στρογγυλοποίηση — για μεγαλύτερη ακρίβεια χρησιμοποιείται double.

4) Επιλογή για bonus με switch

```
printf("1. Ναι\n2. Όχι\nΕπέλεξε (1 ή 2): ");
scanf("%d", &bonusChoice);

switch (bonusChoice) {
    case 1:
        finalGrade += 0.5;
        if (finalGrade > 10) finalGrade = 10;
        break;
    case 2:
        break;
    default:
        printf("Μη έγκυρη επιλογή bonus. Δεν θα προστεθεί bonus.\n");
}
```

- Αν επιλέξει 1, προσθέτουμε 0.5 στον τελικό βαθμό.
- Επειδή μέγιστος βαθμός είναι 10, μετά το += κάνουμε if (finalGrade > 10) finalGrade = 10; (clamping).
- case 2 σημαίνει τίποτα να γίνει (όχι bonus).
- default καλύπτει οποιαδήποτε άλλη μη έγκυρη επιλογή — ενημέρωση χρήστη, δεν προσθέτουμε bonus.

Σημεία προσοχής:

- Χρησιμοποιούμε switch επειδή η επιλογή είναι διακριτή (1 ή 2). Μπορεί να γραφεί και με if/else.
- Αν ο χρήστης βάλει 0 ή 3, θα περάσει στο default και δεν θα έχει bonus.

Παράδειγμα:

- Αν finalGrade πριν ήταν 9.8 και ο χρήστης είχε bonus: 9.8 + 0.5 = 10.3 → το κλειδώνουμε σε 10.

5) Τελική ταξινόμηση με if / else if / else

printf("\nΤελικός βαθμός: %.2f\n", finalGrade);

```
if (finalGrade < 5) {
    printf("Αξιολόγηση: Αποτυχία\n");
} else if (finalGrade < 7) {
    printf("Αξιολόγηση: Επαρκής\n");
} else if (finalGrade < 9) {
    printf("Αξιολόγηση: Πολύ καλός\n");
} else {
    printf("Αξιολόγηση: Άριστος\n");
}
```

Λογική:

- Πρώτα ελέγχουμε αν είναι μικρότερο του 5 → Αποτυχία.
- Αν όχι, ελέγχουμε αν είναι <7 → Επαρκής (αυτό καλύπτει 5.00–6.99…).
- Έπειτα <9 → Πολύ καλός (7.00–8.99…).
- Τέλος ό,τι απομένει (9.00–10.00) → Άριστος.

Σημείο προσοχής:

- Χρησιμοποιούμε < για διαστήματα, οπότε τα όρια συμπεριλαμβάνονται σωστά (π.χ. 7.00 θεωρείται Πολύ καλός).

6) Παραδείγματα εκτέλεσης (πλήρη — με αριθμητικούς υπολογισμούς)

Παράδειγμα Α
Είσοδος:

- e1 = 8.0
- e2 = 6.5
- e3 = 7.0
- exam = 9.0
- bonusChoice = 1 (Ναι)

Υπολογισμός:

- base final = 8.0*0.2* + *6.5*0.2 + 7.0*0.2* + *9.0*0.4 = 7.9
- με bonus: 7.9 + 0.5 = 8.4 → δεν ξεπερνά το 10
  Έξοδος:
- Τελικός βαθμός: 8.40
- Αξιολόγηση: Πολύ καλός

Παράδειγμα Β (clamping)
Είσοδος:

- e1 = 10, e2 = 10, e3 = 10, exam = 10, bonusChoice = 1
  Υπολογισμός:
- base final = 10*0.23* + 10*0.4 = 10
- με bonus -> 10 + 0.5 = 10.5 → clamp σε 10
  Έξοδος:
- Τελικός βαθμός: 10.00
- Αξιολόγηση: Άριστος

Παράδειγμα Γ (αποτυχία)
Είσοδος:

- e1 = 3.0, e2 = 4.0, e3 = 5.0, exam = 4.5, bonusChoice = 2
  Υπολογισμός:
- final = 3*0.2* + *4*0.2 + 5*0.2* + *4.5*0.4
  = 0.6 + 0.8 + 1.0 + 1.8 = 4.2
  Έξοδος:
- Τελικός βαθμός: 4.20
- `Αξιολόγηση: Αποτυχία`