

Εισαγωγή στον Προγραμματισμό

Ασκήσεις σε Δομές
Επιλογής και
Επανάληψης

Φροντιστήριο

Αριστείδης Ηλίας

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

1. Το σούπερ μάρκετ AGI έχει θεσπίσει τη χρήση καρτών προς τους πελάτες του, ώστε να τους ανταποδίδει δωροεπιταγές, ανάλογα με τις αγορές που έχουν πραγματοποιήσει κατά τον προηγούμενο μήνα, σύμφωνα με τον παρακάτω πίνακα:

Αγορές (σε ευρώ)	Ποσοστό επιστροφή (%)
0 – 350	2
351 – 1000	5
1001 και άνω	7

Να αναπτυχθεί αλγόριθμος που θα διαβάζει τον κωδικό του πελάτη και το ποσό των αγορών που έχει πραγματοποιηθεί, και θα υπολογίζει και θα εκτυπώνει το ποσό της δωροεπιταγής που δικαιούται. Θεωρούμε ότι όλες οι τιμές που εισάγει ο χρήστης είναι μέσα στις προβλεπόμενες τιμές.

Αλγόριθμος Δωροεπιταγή

Εμφάνισε 'Δώστε τον κωδικό πελάτη και το ποσό των αγορών'

Διάβασε κωδικός, ποσό_αγορών

Αν ποσό_αγορών $\leq$ 350 τότε

 ποσό_δωροεπιταγής $\leftarrow$ 0.02 * ποσό_αγορών

Αλλιώς_αν ποσό_αγορών $\leq$ 1000 τότε

 ποσό_δωροεπιταγής $\leftarrow$ 0.05 * ποσό_αγορών

Αλλιώς

 ποσό_δωροεπιταγής $\leftarrow$ 0.07 * ποσό_αγορών

Τέλος_αν

Εμφάνισε 'Ο πελάτης ', κωδικός, " δικαιούται δωροεπιταγή ποσού ",
ποσό_δωροεπιταγής

Τέλος Δωροεπιταγή

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

1. Το σούπερ μάρκετ AGI έχει θεσπίσει τη χρήση καρτών προς τους πελάτες του, ώστε να τους ανταποδίδει δωροεπιταγές, ανάλογα με τις αγορές που έχουν πραγματοποιήσει κατά τον προηγούμενο μήνα, σύμφωνα με τον παρακάτω πίνακα:

Αγορές (σε ευρώ)	Ποσοστό επιστροφή (%)
0 – 350	2
351 – 1000	5
1001 και άνω	7

Να αναπτυχθεί αλγόριθμος που θα διαβάζει τον κωδικό του πελάτη και το ποσό των αγορών που έχει πραγματοποιηθεί, και θα υπολογίζει και θα εκτυπώνει το ποσό της δωροεπιταγής που δικαιούται. Θεωρούμε ότι όλες οι τιμές που εισάγει ο χρήστης είναι μέσα στις προβλεπόμενες τιμές.

```
int main()
{
    int kodikos;
    float posoagoron, posodoroepitagis;

    printf("Doste ton kodiko pelati:");
    scanf("%d", &kodikos);
    printf("Doste to poso ton agoron:");
    scanf("%f", &posoagoron);
    if(posoagoron<=350){
        posodoroepitagis=posoagoron*0.02;
    } else if(posoagoron<=1000){
        posodoroepitagis=posoagoron*0.05;
    } else {
        posodoroepitagis=posoagoron*0.07;
    }
    printf("O pelatis me kodiko %d dikaioytai doroepitagi posou %f.\n", kodikos,
posodoroepitagis);
    return 0;
}
```


Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

2. Οι αριθμοί μητρώου των φοιτητών ενός πανεπιστημιακού τμήματος δημιουργούνται ως εξής: Ο αριθμός μητρώου είναι εξαψήφιος αριθμός. Το πρώτο ψηφίο είναι ο αριθμός 1, τα επόμενα δύο ψηφία δείχνουν το έτος εισαγωγής από το οποίο τοποθετούνται τα τελευταία 2 ψηφία του έτους και τα τελευταία 3 ψηφία είναι ο αύξων αριθμός εγγραφής του φοιτητή. Για παράδειγμα ο φοιτητής με αύξοντα αριθμό 114134 εισήχθη το έτος 2014-2015 και έχει γραφτεί με σειρά 134. Για την εξέταση σε ένα μάθημα, ο διδάσκων καθηγητής έχει ανακοινώσει την ακόλουθη σειρά εξέτασης:

Έτος εισαγωγής	Αύξων αριθμός εγγραφής	Ημερομηνία εξέτασης
2014-2015	Όλοι	5 Ιουνίου
2015-2016	Όλοι	6 Ιουνίου
2016-2017	Έως 60	7 Ιουνίου
	61 – 100	8 Ιουνίου
	101 και άνω	10 Ιουνίου
Άλλο	Όλοι	15 Ιουνίου

Να αναπτυχθεί αλγόριθμος που θα διαβάζει τον αριθμό μητρώου ενός φοιτητή και να εκτυπώνει την ημερομηνία εξέτασης του.

Αλγόριθμος Εξέταση

```
Εμφάνισε 'Δώστε αριθμό μητρώου'
Διάβασε αριθμός_μητρώου
έτος_εγγραφής ← αριθμός_μητρώου div 1000
έτος_εγγραφής ← έτος_εγγραφής - 100
αύξων_αριθμός ← αριθμός_μητρώου mod 1000
Αν έτος_εγγραφής = 14 τότε
    Εμφάνισε "Εξετάζεστε 5 Ιουνίου"
Αλλιώς_αν έτος_εγγραφής = 15 τότε
    Εμφάνισε "Εξετάζεστε 6 Ιουνίου"
Αλλιώς_αν έτος_εγγραφής = 16 τότε
    Αν αύξων_αριθμός <= 60 τότε
        Εμφάνισε "Εξετάζεστε 7 Ιουνίου"
    Αλλιώς_αν αύξων_αριθμός <= 100 τότε
        Εμφάνισε "Εξετάζεστε 8 Ιουνίου"
    Αλλιώς
        Εμφάνισε "Εξετάζεστε 10 Ιουνίου"
Τέλος_αν
Αλλιώς
    Εμφάνισε "Εξετάζεστε 15 Ιουνίου"
Τέλος_αν
Τέλος Εξέταση
```



Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

2. Οι αριθμοί μητρώου των φοιτητών ενός πανεπιστημιακού τμήματος δημιουργούνται ως εξής: Ο αριθμός μητρώου είναι εξαψήφιος αριθμός. Το πρώτο ψηφίο είναι ο αριθμός 1, τα επόμενα δύο ψηφία δείχνουν το έτος εισαγωγής από το οποίο τοποθετούνται τα τελευταία 2 ψηφία του έτους και τα τελευταία 3 ψηφία είναι ο αύξων αριθμός εγγραφής του φοιτητή. Για παράδειγμα ο φοιτητής με αύξοντα αριθμό 114134 εισήχθη το έτος 2014-2015 και έχει γραφτεί με σειρά 134. Για την εξέταση σε ένα μάθημα, ο διδάσκων καθηγητής έχει ανακοινώσει την ακόλουθη σειρά εξέτασης:

Έτος εισαγωγής	Αύξων αριθμός εγγραφής	Ημερομηνία εξέτασης
2014-2015	Όλοι	5 Ιουνίου
2015-2016	Όλοι	6 Ιουνίου
2016-2017	Έως 60	7 Ιουνίου
	61 – 100	8 Ιουνίου
	101 και άνω	10 Ιουνίου
Άλλο	Όλοι	15 Ιουνίου

Να αναπτυχθεί αλγόριθμος που θα διαβάζει τον αριθμό μητρώου ενός φοιτητή και να εκτυπώνει την ημερομηνία εξέτασης του.

```
#include <stdio.h>
int main()
{
    int arithmosmitroou, etos_eggrafis, afksonarithmos;
    printf("Doste ariyhmo miroou: ");
    scanf("%d", &arithmosmitroou);
    etos_eggrafis=arithmosmitroou/1000;
    etos_eggrafis=etos_eggrafis-100;
    afksonarithmos=arithmosmitroou%1000;
    if(etos_eggrafis=1){
        printf("Exetazeste 5 Iouniou\n");
    } else if(etos_eggrafis=2){
        printf("Exetazeste 6 Iouniou\n");
    } else if(etos_eggrafis=3){
        if(afksonarithmos<=60){
            printf("Exetazeste 7 Iouniou\n");
        } else if(afksonarithmos<=100){
            printf("Exetazeste 8 Iouniou\n");
        } else {
            printf("Exetazeste 10 Iouniou\n");
        }
    } else {
        printf("Exetazeste 15 Iouniou\n");
    }
    return 0;
}
```

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

3. Τα πακέτα ενοικίασης αυτοκινήτων ενός γραφείου φαίνονται στον ακόλουθο πίνακα:

Τύπος Οχήματος	Κυβισμός (κ. εκ.)	Κόστος ανά ημέρα (€)		
		Ημέρες 1-2	Ημέρες 3-6	Ημέρες >=7
Μοτοσικλέτα	Έως 150	50	45	40
	150 και άνω	80	70	60
Αυτοκίνητο	Έως 120	120	110	90
	1200 – 1400	150	140	120
	1400 και άνω	180	170	150

Να αναπτυχθεί αλγόριθμος που θα διαβάζει το είδος του ενοικιαζόμενου οχήματος, τον κυβισμό του και τις επιθυμητές ημέρες ενοικίασης και θα εκτυπώνει το κόστος ενοικίασης. Στο τέλος θα ερωτάται ο πελάτης εάν επιθυμεί να ασφαλίσει το όχημα και εάν απαντήσει καταφατικά θα πρέπει να υπολογίζεται επιβάρυνση 5% στο κόστος ενοικίασης.

ΤΜΗΜΑ
CEID

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

Αλγόριθμος Ενοικίαση

Εμφάνισε 'Δώστε όχημα, κυβισμό και ημέρες ενοικίασης'

Διάβασε όχημα, κυβισμό, ημέρες

Αν όχημα = "Μοτοσικλέτα" τότε

Αν κυβισμός <= 150 τότε

Αν ημέρες <=2 τότε

χρέωση ← 50 * ημέρες

Αλλιώς_αν ημέρες <=6 τότε

χρέωση ← 45 * ημέρες

Αλλιώς

χρέωση ← 40 * ημέρες

Τέλος_αν

Αλλιώς

Αν ημέρες <=2 τότε

χρέωση ← 80 * ημέρες

Αλλιώς_αν ημέρες <=6 τότε

χρέωση ← 70 * ημέρες

Αλλιώς

χρέωση ← 60 * ημέρες

Τέλος_αν

Τέλος_αν

Αλλιώς

Αν κυβισμός <= 1200 τότε

Αν ημέρες <=2 τότε

χρέωση ← 120 * ημέρες

Αλλιώς_αν ημέρες <=6 τότε

χρέωση ← 110 * ημέρες

Αλλιώς

χρέωση ← 90 * ημέρες

Τέλος_αν

Αλλιώς_αν κυβισμός <= 1400 τότε

Αν ημέρες <=2 τότε

χρέωση ← 150 * ημέρες

Αλλιώς_αν ημέρες <=6 τότε

χρέωση ← 140 * ημέρες

Αλλιώς

χρέωση ← 120 * ημέρες

Τέλος_αν

Αλλιώς

Αν ημέρες <=2 τότε

χρέωση ← 180 * ημέρες

Αλλιώς_αν ημέρες <=6 τότε

χρέωση ← 170 * ημέρες

Αλλιώς

χρέωση ← 150 * ημέρες

Τέλος_αν

Τέλος_αν

Τέλος_αν

Διάβασε ασφάλιση

Αν ασφάλιση = "Ναί" τότε

χρέωση ← χρέωση + χρέωση *5 / 100

Τέλος_αν

Εμφάνισε "Η χρέωση είναι ", χρέωση

Τέλος Ενοικίαση

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

```
#include <stdio.h>
int main()
{
    char typosoximatos, apantisi;
    int kibismos, imeres;
    float xreosi;

    printf("Doste typo oximatos (M- Motosikleta, A- Autokinito): ");
    scanf("%c", &typosoximatos);
    printf("Doste kibismo: ");
    scanf("%d", &kibismos);
    printf("Doste hmeres enoikiashs: ");
    scanf("%d", &imeres);

    if(typosoximatos=='M' || typosoximatos=='m'){
        if(kibismos<=150){
            if(imeres<=2){
                xreosi=imeres*50;
            } else if(imeres<=6){
                xreosi=imeres*45;
            } else {
                xreosi=imeres*40;
            }
        } else {
            if(imeres<=2){
                xreosi=imeres*80;
            } else if(imeres<=6){
                xreosi=imeres*70;
            } else {
                xreosi=imeres*60;
            }
        }
    }
}
```


Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

```
} else if(typosoximatos=='A' || typosoximatos=='a'){
    if(kibismos<=1200){
        if(imeres<=2){
            xreosi=imeres*120;
        } else if(imeres<=6){
            xreosi=imeres*110;
        } else {
            xreosi=imeres*90;
        }
    } else if(kibismos<=1400){
        if(imeres<=2){
            xreosi=imeres*150;
        } else if(imeres<=6){
            xreosi=imeres*140;
        } else {
            xreosi=imeres*120;
        }
    } else {
        if(imeres<=2){
            xreosi=imeres*180;
        } else if(imeres<=6){
            xreosi=imeres*170;
        } else {
            xreosi=imeres*150;
        }
    }
}

printf("Thelete asfalisi (Y/N); ");
scanf(" %c", &apantisi);

if(apantisi=='Y' || apantisi=='y'){
    xreosi=xreosi+xreosi*0.05;
}

printf("Η xreosi einai: %f\n", xreosi);
return 0;
}
```

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

```
#include <stdio.h>
int main()
{
    char typosoximatos, apantisi;
    int kibismos, imeres;
    float xreosi;

    printf("Doste typo oximatos (M- Motosikleta, A- Autokinito): ");
    scanf("%c", &typosoximatos);
    printf("Doste kibismo: ");
    scanf("%d", &kibismos);
    printf("Doste hmeres enoikiashs: ");
    scanf("%d", &imeres);

    if(typosoximatos=='M' || typosoximatos=='m'){
        if(kibismos<=150){
            if(imeres<=2){
                xreosi=imeres*50;
            } else if(imeres<=6){
                xreosi=imeres*45;
            } else {
                xreosi=imeres*40;
            }
        } else {
            if(imeres<=2){
                xreosi=imeres*80;
            } else if(imeres<=6){
                xreosi=imeres*70;
            } else {
                xreosi=imeres*60;
            }
        }
    } else if(typosoximatos=='A' || typosoximatos=='a'){
        if(kibismos<=1200){
            if(imeres<=2){
                xreosi=imeres*120;
            } else if(imeres<=6){
                xreosi=imeres*110;
            } else {
                xreosi=imeres*90;
            }
        } else if(kibismos<=1400){
            if(imeres<=2){
                xreosi=imeres*150;
            } else if(imeres<=6){
                xreosi=imeres*140;
            } else {
                xreosi=imeres*120;
            }
        } else {
            if(imeres<=2){
                xreosi=imeres*180;
            } else if(imeres<=6){
                xreosi=imeres*170;
            } else {
                xreosi=imeres*150;
            }
        }
    }

    printf("Thelete asfalisi (Y/N): ");
    scanf(" %c", &apantisi);

    if(apantisi=='Y' || apantisi=='y'){
        xreosi=xreosi+xreosi*0.05;
    }
    printf("H xreosi einai: %f\n", xreosi);
    return 0;
}
```

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

4. Ένα τυπογραφείο χρεώνει κλιμακωτά τους πελάτες του ως εξής: για τα πρώτα 100 βιβλία [1 - 100] προς 2,80 € το ένα, για τα επόμενα 400 βιβλία [101 - 500] προς 2,40 € το ένα, για τα υπόλοιπα βιβλία (περισσότερα από 500) προς 1,50 € το ένα. Να γραφεί αλγόριθμος που θα υπολογίζει συνολικά τι πρέπει να πληρώσει κάποιος που θέλει να εκτυπώσει κάποια βιβλία.

Αλγόριθμος Τυπογραφείο

Εμφάνισε 'Δώστε αριθμό βιβλίων που θέλετε να τυπώσετε'

Διάβασε α

Αν $\alpha \geq 1$ ΚΑΙ $\alpha \leq 100$ τότε

χρέωση $\leftarrow \alpha * 2.80$

Εκτύπωσε 'Το ποσό που απαιτείται:', χρέωση, 'Ευρώ'

Αλλιώς_αν $\alpha > 100$ ΚΑΙ $\alpha \leq 500$ τότε

χρέωση $\leftarrow 100 * 2.80 + (\alpha - 100) * 2.40$

Εκτύπωσε 'Το ποσό που απαιτείται:', χρέωση, 'Ευρώ'

Αλλιώς_αν $\alpha > 500$ τότε

χρέωση $\leftarrow 100 * 2.80 + 400 * 2.40 + (\alpha - 500) * 1.50$

Εκτύπωσε 'Το ποσό που απαιτείται:', χρέωση, 'Ευρώ'

Αλλιώς

Εκτύπωσε 'Μη αποδεκτός αριθμός βιβλίων'

Τέλος_αν

Τέλος Τυπογραφείο

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

4. Ένα τυπογραφείο χρεώνει κλιμακωτά τους πελάτες του ως εξής: για τα πρώτα 100 βιβλία [1 - 100] προς 2,80 € το ένα, για τα επόμενα 400 βιβλία [101 - 500] προς 2,40 € το ένα, για τα υπόλοιπα βιβλία (περισσότερα από 500) προς 1,50 € το ένα. Να γραφεί αλγόριθμος που θα υπολογίζει συνολικά τι πρέπει να πληρώσει κάποιος που θέλει να εκτυπώσει κάποια βιβλία.

```
int main()
{
    int plithosbooks;
    float xreosi;

    printf("Doste to plithos ton biblion pou thelete na typosete: ");
    scanf("%d", &plithosbooks);

    if(plithosbooks>=1 && plithosbooks<=100){
        xreosi=plithosbooks*2.8;
        printf("To poso pou apaitaitai einai %f euro.\n", xreosi);
    } else if(plithosbooks>100 && plithosbooks<=500){
        xreosi=100*2.8+(plithosbooks-100)*2.4;
        printf("To poso pou apaitaitai einai %f euro.\n", xreosi);
    } else if(plithosbooks>500){
        xreosi=100*2.8+400*2.4+(plithosbooks-500)*1.5;
        printf("To poso pou apaitaitai einai %f euro.\n", xreosi);
    } else {
        printf("Mh apodektos ariyhmos biblion.\n");
    }
    return 0;
}
```


Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

1. Ο υδροφόρος ορίζοντας ενός δήμου εξαντλείται και το δημοτικό συμβούλιο αποφάσισε να πραγματοποιήσει νέα γεώτρηση για την υδροδότηση. Τα πάγια έξοδα για την πληρωμή του συνεργείου είναι 3.000€ και περιλαμβάνουν το κόστος για τα πρώτα 100 μέτρα γεώτρησης. Το 101ο μέτρο κοστίζει 0.80 € και για κάθε επιπλέον μέτρο γεώτρησης το κόστος αυξάνεται κατά 11.3%. Ωστόσο, στατιστικές μελέτες στην περιοχή της γεώτρησης δείχνουν πως αν δεν βρεθεί ικανή ποσότητα νερού σε βάθος μέχρι 700 μέτρα, δεν έχει νόημα να προχωρήσει η εργασία. Να αναπτυχθεί αλγόριθμος που:
- Θα υπολογίζει και θα εκτυπώνει το κόστος για γεώτρηση βάθους 700 μέτρων,
 - Θα διαβάσει το διαθέσιμο για την γεώτρηση ποσό και θα εκτυπώνει το βάθος που μπορεί να φτάσει η γεώτρηση.

Αλγόριθμος Γεώτρηση

όριο_βαθους ← 700

συνολικό_κόστος ← 3000

κόστος_μέτρο ← 0.80

ποσοστό_αύξησης ← 11.3 / 100

Για i από 101 μέχρι όριο_βάθους

 συνολικό_κόστος ← συνολικό_κόστος + κόστος_μέτρο

 κόστος_μέτρο ← κόστος_μέτρο + κόστος_μέτρο * ποσοστό_αύξησης

Τέλος_επανάληψης

Εκτύπωσε "Το κόστος για τα 700 μέτρα είναι: ", συνολικό_κόστος

Διάβασε διαθέσιμα_χρήματα

μέτρα ← 100

συνολικό_κόστος ← 3000

κόστος_μέτρο ← 0.80

Όσο συνολικό_κόστος <= διαθέσιμα_χρήματα επανάλαβε

 συνολικό_κόστος ← συνολικό_κόστος + κόστος_μέτρο

 κόστος_μέτρο ← κόστος_μέτρο + κόστος_μέτρο * ποσοστό_αύξησης

 μέτρα ← μέτρα + 1

Τέλος_επανάληψης

Εκτύπωσε "Με τα διαθέσιμα χρήματα, το μέγιστο βάθος είναι: ", μέτρα, " μέτρα"

Τέλος Γεώτρηση

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

1. Ο υδροφόρος ορίζοντας ενός δήμου εξαντλείται και το δημοτικό συμβούλιο αποφάσισε να πραγματοποιήσει νέα γεώτρηση για την υδροδότηση. Τα πάγια έξοδα για την πληρωμή του συνεργείου είναι 3.000€ και περιλαμβάνουν το κόστος για τα πρώτα 100 μέτρα γεώτρησης. Το 101ο μέτρο κοστίζει 0.80 € και για κάθε επιπλέον μέτρο γεώτρησης το κόστος αυξάνεται κατά 11.3%. Ωστόσο, στατιστικές μελέτες στην περιοχή της γεώτρησης δείχνουν πως αν δεν βρεθεί ικανή ποσότητα νερού σε βάθος μέχρι 700 μέτρα, δεν έχει νόημα να προχωρήσει η εργασία. Να αναπτυχθεί αλγόριθμος που:
- Θα υπολογίζει και θα εκτυπώνει το κόστος για γεώτρηση βάθους 700 μέτρων,
 - Θα διαβάζει το διαθέσιμο για την γεώτρηση ποσό και θα εκτυπώνει το βάθος που μπορεί να φτάσει η γεώτρηση.

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    float orio_vathous=700, synoliko_kostos=3000, kostos_ana_metro=0.8, pososto_auksisis=0.113,$  
    int i, metra=100;
```

```
    for(i=101; i<=orio_vathous; i++){  
        synoliko_kostos=synoliko_kostos+kostos_ana_metro;  
        kostos_ana_metro=kostos_ana_metro+kostos_ana_metro*pososto_auksisis;  
    }
```

```
    printf("To kostos gia ta 700 metra einai %f euro.\n", synoliko_kostos);
```

```
    printf("Dose diathesimo poso: \n");
```

```
    scanf("%f", &diathesimo_poso);
```

```
    synoliko_kostos=3000;
```

```
    kostos_ana_metro=0.8;
```

```
    while(synoliko_kostos<=diathesimo_poso){  
        synoliko_kostos=synoliko_kostos+kostos_ana_metro;  
        kostos_ana_metro = kostos_ana_metro + kostos_ana_metro * pososto_auksisis;  
        metra=metra+1;
```

```
    }
```

```
    printf("Me to diathesimo poso, to megisto bathos einai %d metra.\n", metra);
```

```
    return 0;
```

```
}
```

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

2. Να αναπτύξετε αλγόριθμο που θα διαβάζει αριθμούς αγνώστου πλήθους και θα εκτυπώνει το μέσο όρο των θετικών. Η επαναληπτική διαδικασία να τερματίζεται όταν δοθεί ο αριθμός 0.

Αλγόριθμος Μέσος_Όρος_Αριθμών_εναλ

άθροισμα $\leftarrow$ 0

πλήθος $\leftarrow$ 0

Γράψε "Δώσε έναν ακέραιο αριθμό"

Διάβασε αριθμός

Όσο αριθμός $\neq$ 0 επανάλαβε

Αν (αριθμός $>$ 0) τότε

 άθροισμα $\leftarrow$ άθροισμα + αριθμός

 πλήθος $\leftarrow$ πλήθος + 1

Τέλος_Αν

 Γράψε "Δώσε έναν ακέραιο αριθμό"

 Διάβασε αριθμός

Τέλος_Επανάληψης

Αν πλήθος $\neq$ 0 τότε

 μέσος_όρος $\leftarrow$ άθροισμα / πλήθος

 Εκτύπωσε "Τα στοιχεία που διαβάστηκαν είναι ", πλήθος

 Εκτύπωσε "Ο μέσος όρος είναι ", μέσος_όρος

Αλλιώς

 Εκτύπωσε "Τελικά δεν δόθηκε κανένας αριθμός"

Τέλος_Αν

Τέλος Μέσος_Όρος_Αριθμών_εναλ

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

2. Να αναπτύξετε αλγόριθμο που θα διαβάζει αριθμούς αγνώστου πλήθους και θα εκτυπώνει το μέσο όρο των θετικών. Η επαναληπτική διαδικασία να τερματίζεται όταν δοθεί ο αριθμός 0.

```
#include <stdio.h>

int main()
{
    float mesos_oros;
    int athroisma=0, plithos=0, arithmos;
    printf("Dose enan akeraio arithmo (gia termatismo dose 0):");
    scanf("%d", &arithmos);
    while(arithmos!=0){
        if(arithmos>0){
            athroisma=athroisma+arithmos;
            plithos=plithos+1;
        }
        printf("Dose enan akeraio arithmo (gia termatismo dose 0):");
        scanf("%d", &arithmos);
    }
    if(plithos!=0){
        mesos_oros=athroisma/(float)plithos;
        printf("To plithos ton stoixeion pou diabastikan einai %d.\n", plithos);
        printf("O mesos oros tous einai %f.\n", mesos_oros);
    } else {
        printf("De dothike kanenas arithmos");
    }
    return 0;
}
```


Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

3. Να αναπτυχθεί αλγόριθμος που θα διαβάζει άγνωστο πλήθος αριθμών και θα εντοπίζει και εκτυπώνει το ποσοστό αυτών που είναι πολλαπλάσια του 5. Ο αλγόριθμος θα τερματίζεται όταν εισαχθεί ο αριθμός 0.

Αλγόριθμος Πολλαπλάσιου 5

πολλαπλάσια $\leftarrow$ 0

πλήθος $\leftarrow$ 0

Αρχή_Επανάληψης

Διάβασε αριθμός

Αν (αριθμός $\neq$ 0) και (αριθμός mod 5 = 0) τότε

πολλαπλάσια $\leftarrow$ πολλαπλάσια + 1

Τέλος_Αν

Αν (αριθμός $\neq$ 0) τότε

πλήθος $\leftarrow$ πλήθος + 1

Τέλος_Αν

Μέχρις_Ότου (αριθμός $\neq$ 0)

Αν (πλήθος $\neq$ 0) τότε

ποσοστό $\leftarrow$ 100 * πολλαπλάσια / πλήθος

Εκτύπωσε "Τα πολλαπλάσια του 5 ήταν το ", ποσοστό, "% των αριθμών που διαβάστηκαν"

Αλλιώς

Εκτύπωσε «Δεν υπάρχουν αριθμοί»

Τέλος_αν

Τέλος Πολλαπλάσιου 5

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

3. Να αναπτυχθεί αλγόριθμος που θα διαβάζει άγνωστο πλήθος αριθμών και θα εντοπίζει και εκτυπώνει το ποσοστό αυτών που είναι πολλαπλάσια του 5. Ο αλγόριθμος θα τερματίζεται όταν εισαχθεί ο αριθμός 0.

```
#include <stdio.h>

int main()
{
    int pollaplasia=0, arithmos, plithos=0;
    float pososto;
    do{
        printf("Dose enan akeraio thetiko arithmo (gia termatismo dose 0):");
        scanf("%d", &arithmos);
        if(arithmos!=0 && arithmos%5==0){
            pollaplasia=pollaplasia+1;
        }
        if(arithmos!=0){
            plithos=plithos+1;
        }
    } while(arithmos!=0);
    if(plithos!=0){
        pososto=pollaplasia/(float)plithos*100;
        printf("Ta pollaplasia tou 5 htan to %f %% ton arithmon pou diavastikan.\n",
posost$
    } else {
        printf("Den dothikan kauolou arithmoi.\n");
    }
    return 0;
}
```

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

4. Έστω ότι έχουμε το παρακάτω απόσπασμα αλγορίθμου:

$S \leftarrow 0$

Για i από 5 μέχρι 20 με βήμα 3

Διάβασε X

$S \leftarrow S + X$

Τέλος_επανάληψης

Να ξαναγράψετε το παραπάνω απόσπασμα αλγορίθμου χρησιμοποιώντας αντί για την εντολή Για... Τέλος_επανάληψης:

a) την εντολή Όσο...Τέλος_επανάληψης

b) την εντολή Αρχή_επανάληψης...Μέχρις_ότου

a. $S \leftarrow 0$

$i \leftarrow 5$

ΟΣΟ $i \leq 20$ ΕΠΑΝΑΛΑΒΕ

Διάβασε x

$S \leftarrow S + x$

$i \leftarrow i + 3$

Τέλος_επανάληψης

b. $S \leftarrow 0$

$i \leftarrow 5$

Αρχή_Επανάληψης

Διάβασε x

$S \leftarrow S + x$

$i \leftarrow i + 3$

μέχρις_ότου $i > 20$

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

4. Έστω ότι έχουμε το παρακάτω απόσπασμα αλγορίθμου:

$S \leftarrow 0$

Για i από 5 μέχρι 20 με βήμα 3

Διάβασε X

$S \leftarrow S + X$

Τέλος_επανάληψης

Να ξαναγράψετε το παραπάνω απόσπασμα αλγορίθμου χρησιμοποιώντας αντί για την εντολή Για... Τέλος_επανάληψης:

a) την εντολή Όσο...Τέλος_επανάληψης

b) την εντολή Αρχή_επανάληψης...Μέχρις_ότου

a. $S=0;$

$i=5;$

```
while (i <= 20) {  
    printf("%d", &x);  
    S=S + x;  
    I=i + 3;  
}
```

b. $S=0;$

$i=5;$

do{

```
    printf("%d", &x);
```

```
    S=S + x;
```

```
    I=i + 3;
```

```
}while (i>20);
```


Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

5. Σε κάποιο αγώνα στίβου, για το άθλημα «Άλμα εις μήκος» καταγράφεται για κάθε αθλητή η καλύτερη έγκυρη επίδοσή του. Τιμής ένεκεν, πρώτος αγωνίζεται ο περσινός πρωταθλητής. Η επιτροπή του αγώνα διαχειρίζεται τα στοιχεία των αθλητών που αγωνίστηκαν.

Να γράψετε πρόγραμμα το οποίο:

- a) Να ζητάει το ρεκόρ αγώνων και να το δέχεται, εφόσον είναι θετικό και μικρότερο των 9 μέτρων.
- b) Να ζητάει τον συνολικό αριθμό των αγωνιζομένων και για κάθε αθλητή τον τριψήφιο αριθμό συμμετοχής και την επίδοσή του σε μέτρα με τη σειρά που αγωνίστηκε.
- c) Να εμφανίζει τον τριψήφιο αριθμό συμμετοχής του αθλητή με τη χειρότερη επίδοση.
- d) Να εμφανίζει του τριψήφιους αριθμούς συμμετοχής των αθλητών που κατέρριψαν το ρεκόρ αγώνων. Αν δεν υπάρχουν τέτοιοι αθλητές, να εμφανίζει το πλήθος των αθλητών που πλησίασαν το ρεκόρ αγώνων σε απόσταση όχι μεγαλύτερη των 50 εκατοστών.
- e) Να βρίσκει και να εμφανίζει τη θέση που κατέλαβε στην τελική κατάταξη ο περσινός πρωταθλητής.

Σημείωση: Να θεωρήσετε ότι κάθε αθλητής έχει έγκυρη επίδοση και ότι όλες οι επιδόσεις των αθλητών που καταγράφονται είναι διαφορετικές μεταξύ τους.

ΤΜΗΜΑ
CEID

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

Αλγόριθμος Άλμα

Κατερριψαντορεκορ $\leftarrow 0$

Πλησιασανστα50 $\leftarrow 0$

Αρχή_επανάληψης

Γράψε «Δώσε το Ρεκόρ»

Διάβασε Ρεκόρ

Μέχρις_ότου (Ρεκόρ < 0 ή Ρεκόρ > 9)

Εμφάνισε "Δώσε συνολικό αριθμό αγωνιζομένων"

Διάβασε πλήθοςαγωνιζομένων

Αρχή_επανάληψης

Εμφάνισε "Δώσε αριθμό περσινού πρωταθλητή"

Διάβασε αριθμόσυμπρωταθλητή

Μέχρις_ότου (αριθμόσυμπρωταθλητή <100 ή αριθμόσυμπρωταθλητή >1000)

Εμφάνισε "Δώσε επίδοση περσινού πρωταθλητή"

Διάβασε επιδοσηπρωταθλ

minεπιδοση $\leftarrow$ επιδοσηπρωταθλ

minαριθμσυμμετοχ $\leftarrow 1$

θέσηπρωταθλ $\leftarrow 1$

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

Για i από 2 μέχρι $plithathloto$ Με_βήμα 1

Αρχή_Επανάληψης

Γράψε "Δώσε αριθμό συμμετοχής αθλητή:"

Διάβασε αριθμσυμμαθλητη

Μέχρις_ότου (αριθμσυμμαθλητη<100 ή αριθμσυμμαθλητη>1000)

Γράψε "Δώσε επίδοση αθλητή"

Διάβασε επίδοση

Αν (minεπίδοση>επίδοση) τότε

minεπίδοση=επίδοση

minαριθμσυμμετοχ=αριθμσυμμαθλητη

τέλος_αν

Αν (επίδοση>ρεκόρ) τότε

Γράψε "Ο αθλητής με αριθμό ", αριθμσυμμαθλητη, κατέρριψε το ρεκόρ"

Κατερριψαντορεκορ= Κατερριψαντορεκορ +1

Τέλος_αν

Αν ((ρεκόρ-επίδοση)>0.5) τότε

Πλησιασανστα50= Πλησιασανστα50+1

Τέλος_αν

Αν (επίδοση>επιδοσηπρωταθλ) τότε

θέσηπρωταθλ=θέσηπρωταθλ+1

Τέλος_αν

Τέλος_επανάληψης

Αν (Κατερριψαντορεκορ=0) τότε

Γράψε "Δεν κατέρριψε κανένας το ρεκόρ, αλλά ", Πλησιασανστα50, " αθλητές το πλησίασαν"

Τέλος_αν

Γράψε "Ο περσινός πρωταθλητής κατετάγη στην ", θέσηπρωταθλ

Τέλος Άλμα

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

Αλγόριθμος Άλμα

```
Κατερριψαντορεκορ ← 0
Πλησιασανστα50 ← 0
Αρχή_επανάληψης
    Γράψε «Δώσε το Ρεκόρ»
    Διάβασε Ρεκόρ
Μέχρις_ότου (Ρεκόρ < 0 ή Ρεκόρ > 9 )
Εμφάνισε "Δώσε συνολικό αριθμό αγωνιζομένων"
Διάβασε πλήθοςαγωνιζομένων
Αρχή_επανάληψης
    Εμφάνισε "Δώσε αριθμό περσινού πρωταθλητή"
    Διάβασε αριθμόσυμπρωταθλητή
Μέχρις_ότου (αριθμόσυμπρωταθλητή<100 ή αριθμόσυμπρωταθλητή>1000)
Εμφάνισε "Δώσε επίδοση περσινού πρωταθλητή"
Διάβασε επιδοσηπρωταθλ
minεπιδοση ← επιδοσηπρωταθλ
minαριθμσυμμετοχ←1
θέσηπρωταθλ←1
Για i από 2 μέχρι plithathloton Με_βήμα 1
    Αρχή_Επανάληψης
        Γράψε "Δώσε αριθμό συμμετοχής αθλητή:"
        Διάβασε αριθμσυμμαθλητη
Μέχρις_ότου (αριθμσυμμαθλητη<100 ή αριθμσυμμαθλητη>1000)
        Γράψε "Δώσε επίδοση αθλητή"
        Διάβασε επίδοση
        Αν (minεπιδοση>επίδοση) τότε
            minεπίδοση=επίδοση
            minαριθμσυμμετοχ=αριθμσυμμαθλητη
        τέλος_αν
        Αν (επίδοση>ρεκόρ) τότε
            Γράψε "Ο αθλητής με αριθμό ", αριθμσυμμαθλητη, κατέρριψε το ρεκόρ"
            Κατερριψαντορεκορ= Κατερριψαντορεκορ +1
        Τέλος_αν
        Αν ((ρεκόρ-επίδοση)>0.5) τότε
            Πλησιασανστα50= Πλησιασανστα50+1
        Τέλος_αν
        Αν (επίδοση>επιδοσηπρωταθλ) τότε
            θέσηπρωταθλ=θέσηπρωταθλ+1
        Τέλος_αν
    Τέλος_επανάληψης
Αν (Κατερριψαντορεκορ=0) τότε
    Γράψε "Δεν κατέρριψε κανένας το ρεκόρ, αλλά ", Πλησιασανστα50, " αθλητές το πλησίασαν"
Τέλος_αν
Γράψε "Ο περσινός πρωταθλητής κατετάγη στην ", θέσηπρωταθλ
Τέλος Άλμα
```


Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

```
#include <stdio.h>
int main()
{
    float rekor, epidosiprotathliti, epidosi, minepidosis;
    int i, plithathlonton, arithmossymmetoxis, arithmsymmin, katerripsanrekor=0,
    plisiasanta50=0$
    do{
        printf("Dose to rekor agonon:");
        scanf("%f", &rekor);
    }while(rekor<0 || rekor>9);
    printf("Posoi einai oi agonizomenoi:");
    scanf("%d", &plithathlonton);
    do{
        printf("Dose arithmo symmetoxis persinou protathliti:");
        scanf("%d", &arithmossymmetoxis);
    } while(arithmossymmetoxis<100 || arithmossymmetoxis>1000);
    printf("Dose epidosi persinou protathliti:");
    scanf("%f", &epidosiprotathliti);
    minepidosis=epidosiprotathliti;
    arithmsymmin=arithmossymmetoxis;
```

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

```
for(i=1; i<plithathlonton; i++){
    do{
        printf("Dose arithmo symmetoxis athliti:");
        scanf("%d", &arithmossymmetoxis);
    } while(arithmossymmetoxis<100 || arithmossymmetoxis>1000);
    printf("Dose epidosi athliti:");
    scanf("%f", &epidosi);
    if(minepidosis>epidosi){
        minepidosis=epidosi;
        minepidosis=epidosi;
        arithmsymmin=arithmossymmetoxis;
    }
    if(epidosi>rekor){
        printf("O athlitis me arithmo %d katerripse to rekor.\n", arithmossymmetoxi$
        katerripsanrekor=katerripsanrekor+1;
    }
    if((rekor-epidosi)>0.5){
        plisiasanta50=plisiasanta50+1;
    }
    if(epidosi>epidosiprotathliti){
        thesiprot=thesiprot+1;
    }
}
if(katerripsanrekor==0){
    printf("Den katerripse kanenas to recor, alla %d athlites to plisiasan.\n", plisiasanta50);
}
printf("O persinos protathlitis katetagei stin %d thesi\n.", thesiprot);
return 0;
}
```

Προτεινόμενες Ασκήσεις (Μη Βαθμολογούμενες)

```
#include <stdio.h>
int main()
{
    float rekor, epidosi, minepidosis;
    int i, plithathloto, arithmosymmetoxis, arithmsymmin, katerripsanrekor=0, plisiasanta50=0;
    do{
        printf("Dose to rekor agonon:");
        scanf("%f", &rekor);
    }while(rekor<0 || rekor>9);
    printf("Posoi einai oi agonizomenoi:");
    scanf("%d", &plithathloto);
    do{
        printf("Dose arithmo symmetoxis persinou protathliti:");
        scanf("%d", &arithmosymmetoxis);
    } while(arithmosymmetoxis<100 || arithmosymmetoxis>1000);
    printf("Dose epidosi persinou protathliti:");
    scanf("%f", &epidosi);
    minepidosis=epidosi;
    arithmsymmin=arithmosymmetoxis;
    for(i=1; i<plithathloto; i++){
        do{
            printf("Dose arithmo symmetoxis athliti:");
            scanf("%d", &arithmosymmetoxis);
        } while(arithmosymmetoxis<100 || arithmosymmetoxis>1000);
        printf("Dose epidosi athliti:");
        scanf("%f", &epidosi);
        if(minepidosis>epidosi){
            minepidosis=epidosi;
            minepidosis=epidosi;
            arithmsymmin=arithmosymmetoxis;
        }
        if(epidosi>rekor){
            printf("O athlitis me arithmo %d katerripse to rekor.\n", arithmosymmetoxis);
            katerripsanrekor=katerripsanrekor+1;
        }
        if((rekor-epidosi)>0.5){
            plisiasanta50=plisiasanta50+1;
        }
        if(epidosi>epidosi){
            thesiproto=thesiproto+1;
        }
    }
    if(katerripsanrekor==0){
        printf("Den katerripse kanenas to rekor, alla %d athlites to plisiasan.\n", plisiasanta50);
    }
    printf("O persinos protathlitis katetagei stin %d thesi\n.", thesiproto);
    return 0;
}
```


Ευχαριστώ

Ερωτήσεις?