

1 ΣΕΙΡΑ

1. Να βρείτε και να διορθώσετε τα συντακτικά λάθη στα πιο κάτω προγράμματα :

1. include <stdio.h>	1. #include <stdio.h>
2. int main(void){	2. int main(void)
3. int i	3. {
4. i:=66;	4. /* Πως σε λένε;
5. printf("i=%d\n", i);	5. printf "Με λένε Πόπη.\n";
6. ;	6. return 0;
7. return "0";	7. }
8. }	

2. Να βρείτε και να διορθώσετε τα λάθη στο πιο κάτω πρόγραμμα :

```
1. #include <stdio.h>
2. INT main(Void)
3. {
4. INT sum = 0.0;
5. sum = 5 + 3 // compute //
6. printf("Result is %d\n" sum);
7. return 666;
8. }
```

3. Να εντοπίσετε τα λάθη που υπάρχουν στον πιο κάτω κώδικα της C. Αφού τον διορθώσετε να τον εκτελέσετε και να βρείτε τι εμφανίζετε στην οθόνη.

```
1. #include <stdio.h>
2. #define MAX = 2010;
3. int main(void)
4. {
5. int a = 3, b = 4, c;
6. c == a + b;
7. MAX = 1969;
8. printf("a + b = %d and MAX is now %d\n", c, MAX);
9. return 0;
10. }
```

4. Να σχεδιάσετε και να δημιουργήσετε πρόγραμμα σε C το οποίο να διαβάζει από το πληκτρολόγιο δύο ακέραιους αριθμούς τους οποίους καταχωρεί σε δυο κατάλληλες μεταβλητές Num1 και Num2. Στη συνέχεια ανταλλάσει το περιεχόμενο των μεταβλητών αυτών. Για επιβεβαίωση της ορθότητας, να σχηματίζει και να εμφανίζει στην οθόνη τη διαφορά των μεταβλητών πριν και μετά την ανταλλαγή του περιεχομένου τους. Η εμφάνιση των αποτελεσμάτων θα έχει για παράδειγμα τη μορφή Num1=20 Num2=10 Num1-Num2=10 στην επόμενη

γραμμή, μετά την ανταλλαγή $\text{Num1}=10$ $\text{Num2}=20$ $\text{Num1}-\text{Num2}=-10$ Όπως καταλαβαίνετε τα δυο αποτελέσματα πρέπει να είναι αντίθετοι αριθμοί.

5. Να σχεδιάσετε και να δημιουργήσετε πρόγραμμα σε C το οποίο θα ζητά από το πληκτρολόγιο τις τιμές δύο ωμικών αντιστάσεων $R1$ και $R2$, το είδος της σύνδεσής τους, αν είναι σε σειρά ή παράλληλα. Αν είναι σε σειρά θα διαβάζει την τιμή 1 ενώ αν είναι παράλληλα θα διαβάζει την τιμή 2. Στη συνέχεια ανάλογα με το είδος της σύνδεσης θα υπολογίζει και θα εμφανίζει τη συνολικά αντίσταση R του κυκλώματος. (Υπενθυμίζεται ότι η $R = R1 + R2$ για σύνδεση σε σειρά ενώ $1/R = 1/R1 + 1/R2$ για παράλληλη σύνδεση).
6. Ένα έτος είναι δίσεκτο όταν διαιρείται ακριβώς με το 4 και το 400. Εξαιρούνται τα έτη που διαιρούνται ακριβώς με το 100. Δηλαδή το 1900 διαιρείται ακριβώς με το 4 και δεν διαιρείται ακριβώς με το 400, διαιρείται όμως ακριβώς με το 100. Για αυτό δεν είναι δίσεκτο. Ενώ το 2000 ήταν, γιατί διαιρείται μεν με το 100 αλλά διαιρείται ακριβώς και με το 400. Τα δίσεκτα έτη έχουν 366 ημέρες, μία περισσότερη από ότι τα κανονικά έτη που έχουν 365. Η πρόσθετη αυτή ημέρα καταλογίζεται στον Φεβρουάριο που τα δίσεκτα έτη έχουν 29 ημέρες αντί 28. Να σχεδιάσετε και να δημιουργήσετε πρόγραμμα σε C το οποίο θα ζητά να του καταχωρηθεί μια ημερομηνία γέννησης και η σημερινή ημερομηνία με τρεις ακέραιους αριθμούς η κάθε μια. Δηλαδή για παράδειγμα, για την ημερομηνία γέννησης θα ορίσετε τρεις ακέραιες μεταβλητές BDay, BMonth και BYear και θα πληκτρολογήσετε με τη σειρά, για κάθε μία από αυτές, 17, 4 και 2003 που θα σημαίνει ημερομηνία γέννησης 17/4/2003. Το πρόγραμμα να απαντά πόσες συνολικά ημέρες έχουν περάσει από αυτή. Θα χρειαστεί να εφοδιάσετε το πρόγραμμά σας με την πληροφορία του πόσες ημέρες έχει κάθε μήνας. Για τον υπολογισμό των ημερών θα υπολογίσετε πόσες ημέρες απομένουν μέχρι το τέλος της χρονιάς γέννησης, πόσες ημέρες πέρασαν ήδη τη φετινή χρονιά, και πόσες είναι συνολικά οι ημέρες τα μεσολαβούντα ακέραια έτη. Το άθροισμα αυτών θα σας δώσει τις συνολικές ημέρες που μεσολάβησαν. Θα εκτιμηθεί αν εφοδιάσετε το πρόγραμμα με ελέγχους εγκυρότητας π.χ. να διαπιστώνει για παράδειγμα ότι δεν υπάρχει ημερομηνία 32/7/2003 ή 27/13/2003 κλπ.
7. Να σχεδιάσετε και να δημιουργήσετε πρόγραμμα σε C το οποίο να ζητά και να διαβάζει ακέραιο αριθμό, μεγαλύτερο το 10, από το πληκτρολόγιο. Να εξετάζει στη συνέχεια αν ο αριθμός αυτός είναι καρκινικός (παλίνδρομος) ή όχι. Καρκινικός λέγεται ο αριθμός που διαβάζεται το ίδιο και από το τέλος προς την αρχή. Για παράδειγμα ο 1234321 είναι καρκινικός ενώ ο 1234123 όχι. Το πρόγραμμα να εμφανίζει μήνυμα στην οθόνη ότι ο δεδομένος αριθμός είναι ή δεν είναι καρκινικός.

8. Να σχεδιάσετε και να δημιουργήσετε πρόγραμμα το οποίο θα ζητά από το χρήστη με τη σειρά δύο πραγματικούς γενικά αριθμούς, και ένα χαρακτήρα. Ανάλογα με τον χαρακτήρα που πληκτρολογήθηκε θα εκτελεί με τους αριθμούς την αντίστοιχη αριθμητική πράξη. Αν πληκτρολογηθεί ο χαρακτήρας + θα εκτελεί πρόσθεση, για - αφαίρεση, για x πολλαπλασιασμό και για : διαίρεση. Αν πληκτρολογηθεί οποιοσδήποτε άλλος χαρακτήρας θα εμφανίζει μήνυμα ότι δεν είναι αναγνωρίσιμη πράξη και θα ζητά ξανά να πληκτρολογηθεί νέος σωστός χαρακτήρας. Το πρόγραμμα θα εμφανίζει στην οθόνη τους δύο αριθμούς έχοντας ενδιάμεσα τον χαρακτήρα που πληκτρολογήθηκε, θα ακολουθεί το = με το αποτέλεσμα της πράξης. Το πρόγραμμα θα τερματίζεται αν δοθούν ως αριθμοί δύο μηδενικά. Θα εκτιμηθεί αν εφοδιάσετε το πρόγραμμα με ελέγχους εγκυρότητας.
9. Να σχεδιαστεί και να συνταχθεί πρόγραμμα σε C το οποίο να δέχεται έναν ακέραιο θετικό αριθμό από το πληκτρολόγιο και να εξετάζει αν ο αριθμός είναι πρώτος ή όχι. Πρώτος λέγεται ο ακέραιος που διαιρείται ακριβώς με τη μονάδα και τον εαυτό του μόνο. Επομένως ο αριθμός πρέπει να είναι μονός, (περιττός) και θα εξετάζονται μόνο μονοί (περιττοί) διαιρέτες. Προσπαθήστε να βελτιώσετε όσο περισσότερο μπορείτε τον αλγόριθμο και κατά συνέπεια το πρόγραμμά σας.
10. Να σχεδιαστεί και να συνταχθεί πρόγραμμα σε C το οποίο να δέχεται έναν ακέραιο θετικό αριθμό από το πληκτρολόγιο και να εξετάζει αν ο αριθμός είναι τέλειος. Τέλειος λέγεται ο ακέραιος του οποίου οι διαιρέτες έχουν άθροισμα τον ίδιο τον αριθμό. Για παράδειγμα ο αριθμός 28 έχει ως διαιρέτες το (1, 2, 4, 7, 14) το άθροισμά τους είναι $1+2+4+7+14=28$ επομένως ο 28 είναι τέλειος σε αντίθεση με το 32 του οποίου οι διαιρέτες είναι (1, 2, 4, 8, 16) και $1+2+4+8+16=31$ δεν ταυτίζεται με το 32.