

1. Να γράψετε μια συνάρτηση η οποία υπολογίζει και επιστρέφει (εμμέσως) την τιμή της μαθηματικής παράστασης  $x^3+5x^2+8.1x+9.6$  για οποιοδήποτε ακέραιο  $x$ . (δεύτερη και τρίτη παραλλαγή, η πρώτη υλοποιήθηκε στο Εργ.2 )

```
// 2.
```

```
void fmath(int x, float &y );
```

```
// κληση
```

```
float g;
```

```
fmath(4,g);
```

```
// 3.
```

```
void fmath(int x, float *y); // πρωτοτυπο
```

```
// κληση
```

```
float g;
```

```
fmath(4,&g);
```

2. Να γράψετε μια **char** συνάρτηση η οποία ελέγχει έναν χαρακτήρα. Εάν χαρακτήρας είναι κεφαλαίο γράμμα (Α έως Ζ) τότε μετατρέπεται στον αντίστοιχο πεζό χαρακτήρα (η συνάρτηση επιστρέφει αυτόν τον πεζό χαρακτήρα). Εάν ο χαρακτήρας δεν είναι κεφαλαίο γράμμα τότε η συνάρτηση επιστρέφει τον χαρακτήρα ως έχει.

```
char ksp(char a);
```

Παραλλαγή 1: void ksp(char a, char &b);

Παραλλαγή 2: void ksp(char &a);

(Ποιες είναι η διαφορές μεταξύ 1ης και 2ης παραλλαγής;

Παραλλαγή 3: void ksp(char a, char \*b);

3. Να γράψετε μια **void** συνάρτηση η οποία ελέγχει έναν ακέραιο αριθμό και επιστρέφει “εμμέσως” (έχει ως “έξοδο”) α) την απόλυτη τιμή του αριθμού, β) μια σημαία-χαρακτήρα (flag) με τιμή ‘Y’ εάν ο αριθμός διαιρείται ακριβώς δια του 3, εάν όχι με τιμή ‘N’.

```
void abx(int x, int &a, char &flag);
```

Παραλλαγή: Με χρήση δεικτών

```
void abx(int x, int *a, char *flag);
```

4. Να γράψετε δυο συναρτήσεις με το όνομα **embadon** όπου υπολογίζεται το εμβαδόν ενός τετράγωνου (κλήση με μια παράμετρο) ή το εμβαδόν ενός ορθογωνίου παραλληλεπίπεδου (κλήση με δυο παραμέτρους – μήκος, πλάτος). Να χρησιμοποιηθεί υπερφόρτωση συναρτήσεων.

```
float embadon(float m);
```

```
float embadon(float m, float p);
```

5. Να γράψετε ένα πρόγραμμα με δυο συναρτήσεις:

- Η 1η συνάρτηση με πρωτότυπο **void diav(int a[], int n)** θα εισάγει από το πληκτρολόγιο (τερματικό) ακέραιους σε έναν πίνακα ακεραίων **a** με **n** στοιχεία.

- Η 2η συνάρτηση **float moa(int a[], int n1, int n2)** επιστρέφει τον μέσο όρο των αριθμών του πίνακα **a** από το κελί **n1** έως και το κελί **n2**.

- Στο κυρίως πρόγραμμα δηλώσετε (στατικά) έναν πίνακα ακεραίων 10 στοιχείων. Ο **n1** και **n2** δίδονται από το τερματικό. Να γίνεται έλεγχος: 1) ώστε ο **n2** να μη είναι μεγαλύτερος από το πλήθος **n** των κελιών του πίνακα, 2) ο **n1** να μη είναι μεγαλύτερος του **n2**.

6. Η άσκηση 5. με χρήση δεικτών (αριθμητική δεικτών)

```
void diav(int *a, int n);  
float moa(int *a, int n1, int n2);
```

7. Να γραψετε ενα προγραμμα οπου στο κυρίως πρόγραμμα: i) το οποιο διαβαζει απο το πληκτρολογιο με **cin** έναν ακέραιο αριθμό **n** που αντιπροσωπευει «έγκυρες» μεσημεριανές θερμοκρασίες. ii) δημιουργήσετε ένα πίνακα με **n** στοιχεία.

Επισης γράψετε

A) μια **float** συνάρτηση που καταχωρεί **n** έγκυρες θερμοκρασίες στον πίνακα. Οι θερμοκρασίες δημιουργούνται τυχαία και κυμαίνονται από -60.0 έως +40.0 αλλά οι εγκυρες θερμοκρασίες κυμαίνονται από -20.0 έως +30.0. Η συναρτηση επιστρέφει το πλήθος των μη-έγκυρων θερμοκρασιών.

```
float diav(int a[], int n);
```

B) μια **float** συνάρτηση που επιστρεφει την μεση θερμοκρασία των θερμοκρασιών του πίνακα.

```
float methe(int a[], int n);
```

Γ) μια **void** συνάρτηση που επιστρεφει «εμμεσα» α) το πληθος των ζεστων ημερων (θερμοκρασιες μεγαλυτερες από 17 εως και 28), β) Την μεση θερμοκρασια των κρυων ημερων (θερμοκρασιες απο 1 εως και 17).

```
void zekr(int a[], int n, int &pze, float *mkr);
```

Δ) μια **int** συνάρτηση που επιστρεφει το πληθος των ιδανικων ημερων με θερμοκρασια 22.

```
int perf(int a[], int n);
```

Ε) μια συνάρτηση που επιστρεφει έναν **int** δεικτη (δευθυνση μνήμης) ο οποίος δείχνει στην πιο ζεστή θερμοκρασία από τις θερμοκρασίας που δοθήκαν.

```
int* ptomx(int a[], int n);
```

ΣΤ) μια **short** συνάρτηση που επιστρεφει την θεση (πχ. 5<sup>n</sup> ή 20<sup>n</sup> ή 1234<sup>n</sup>) στον πίνακα δηλ. σε πιο κελι του πίνακα βρίσκεται η πιο κρύα θερμοκρασία.

Στο κυρίως προγραμμα να εμφανισθουν στην οθονη με **cout**

α. Το πλήθος των μη-εγκυρων θερμοκρασιών.

β. Η μεση θερμοκρασια των θερμοκρασιών.

γ. Το πληθος των ζεστων ημερων (θερμοκρασιες μεγαλυτερες απο 17 εως και 28).

δ. Την μεση θερμοκρασια των κρυων ημερων (θερμοκρασιες απο 1 εως και 17)

ε. Το πληθος των ιδανικων ημερων με θερμοκρασια 22.

στ. Την διευθυνση μνήμης που βρισκεται η πιο ζεστή θερμοκρασία και την θερμοκρασία της πιο ζέστης ημερας.

ζ. Την θεση (κελλι) που βρισκεται η πιο κρύα θερμοκρασία και την τιμή της πιο κρύας ημερας.

8. Να γράψετε: Ένα πρόγραμμα όπου, στο κυρίως πρόγραμμα δηλώσετε δυο πίνακες 4x3, έναν πίνακα **double x** και έναν πίνακα ακεραίων **y**.

- Μια **void** συνάρτηση η οποία διαβάζει αριθμούς από το τερματικό για τον πίνακα **x**.

- Στην συνέχεια μια άλλη **void** συνάρτηση επεξεργάζεται τον πίνακα **x** και τα αποτελέσματα καταχωρούνται στον πίνακα **y**. Στην θέση (κελλι) του πίνακα **x** όπου υπάρχει θετικός αριθμός καταχωρείται ο αριθμός 1 στο αντίστοιχο κελί του **y**, στην θέση που υπάρχει αρνητικός αριθμός καταχωρείται το -1, και στην θέση που υπάρχει 0 (μηδέν) καταχωρείται το 0 (μηδέν).

- Τέλος μια συνάρτηση εμφανίζει τον πίνακα **y** και επιστρέφει το πλήθος των μη μηδενικών στοιχείων του πίνακα **y**.

```
void eis(double x[][3]);  
void ant(double x[][3],int y[][3]);  
int emf(int y[][3]);
```

E03\_10-11X