

ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ, ΑΘΛΗΤΙΣΜΟΥ ΚΑΙ ΝΕΟΛΑΙΑΣ
ΔΙΕΥΘΥΝΣΗ ΜΕΣΗΣ ΤΕΧΝΙΚΗΣ ΚΑΙ ΕΠΑΓΓΕΛΜΑΤΙΚΗΣ ΕΚΠΑΙΔΕΥΣΗΣ
ΚΑΙ ΚΑΤΑΡΤΙΣΗΣ

ΕΝΙΑΙΕΣ ΓΡΑΠΤΕΣ ΤΕΛΙΚΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΚΑΙ
ΑΠΟΛΥΤΗΡΙΕΣ ΕΞΕΤΑΣΕΙΣ

20 23 - 20 24

Γ΄ ΤΑΞΗΣ ΤΕΣΕΚ

ΣΕΙΡΑ Α΄

ΘΕΩΡΗΤΙΚΗ ΚΑΤΕΥΘΥΝΣΗ

ΗΜΕΡΟΜΗΝΙΑ : Παρασκευή, 24 Μαΐου 2024

ΕΞΕΤΑΖΟΜΕΝΟ ΜΑΘΗΜΑ: Εφαρμογές Προγραμματισμού III-TEM2

ΚΩΔΙΚΟΣ ΜΑΘΗΜΑΤΟΣ : thips302

ΣΥΝΟΛΙΚΗ ΔΙΑΡΚΕΙΑ ΓΡΑΠΤΗΣ ΕΞΕΤΑΣΗΣ: 90΄ λεπτά

ΤΟ ΕΞΕΤΑΣΤΙΚΟ ΔΟΚΙΜΙΟ ΑΠΟΤΕΛΕΙΤΑΙ ΑΠΟ ΔΕΚΑΤΕΣΣΕΡΙΣ (14) ΣΕΛΙΔΕΣ.

ΤΑ ΜΕΡΗ ΤΟΥ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ ΕΙΝΑΙ ΤΡΙΑ (Α΄, Β΄ ΚΑΙ Γ΄).

ΟΔΗΓΙΕΣ (για τους εξεταζόμενους)

1. Να απαντήσετε **ΟΛΑ** τα ερωτήματα πάνω στο εξεταστικό δοκίμιο.
2. Να μη γράψετε πουθενά το όνομα σας στο εξεταστικό δοκίμιο εκτός του καθορισμένου χώρου στο χαρτονάκι που σας έχει δοθεί.
3. Να απαντήσετε σε όλα τα θέματα μόνο με πένα χρώματος μπλε ανεξίτηλης μελάνης. Μολύβι επιτρέπεται, μόνο αν το ζητάει η εκφώνηση, και μόνο για σχήματα, πίνακες, διαγράμματα κλπ.
4. Απαγορεύεται η χρήση διορθωτικού υγρού και διορθωτικής ταινίας.
5. Επιτρέπεται η χρήση μη προγραμματιζόμενης υπολογιστικής μηχανής.

ΣΑΣ ΕΥΧΟΜΑΣΤΕ ΚΑΛΗ ΕΠΙΤΥΧΙΑ

ΟΔΗΓΙΕΣ (για την επιτροπή εξετάσεων)

1. Το εξεταστικό δοκίμιο να εκτυπωθεί και στις δύο όψεις.

ΤΟ ΕΞΕΤΑΣΤΙΚΟ ΔΟΚΙΜΙΟ ΝΑ ΕΚΤΥΠΩΘΕΙ: ΜΑΥΡΟΑΣΠΡΟ

ΜΕΡΟΣ Α΄: Αποτελείται από έξι (6) ερωτήσεις. Κάθε ορθή απάντηση βαθμολογείται με οκτώ (8) μονάδες.

1. Να χαρακτηρίσετε με σωστό (Σ) ή λάθος (Λ) καθεμιά από τις παρακάτω προτάσεις. Όλες οι προτάσεις αφορούν τη γλώσσα προγραμματισμού C++.

A/A	Πρόταση	Σωστό / Λάθος
α)	Ο μέγιστος αριθμός των συγκρίσεων στη σειριακή αναζήτηση (sequential search) είναι ίσος με το πλήθος των στοιχείων του πίνακα.	
β)	Η δυαδική αναζήτηση (binary search) μπορεί να χρησιμοποιηθεί μόνο αν ο πίνακας είναι ήδη ταξινομημένος.	
γ)	Ο αλγόριθμος της δυαδικής αναζήτησης (binary search) ξεκινά εξετάζοντας το «πρώτο» στοιχείο του πίνακα.	
δ)	Ο αλγόριθμος της φυσαλίδας (bubble sort) ταξινομεί τα στοιχεία, χρησιμοποιώντας την αρχή της σύγκρισης και αντιμετάθεσης δύο γειτονικών στοιχείων.	
ε)	Ο αλγόριθμος φυσαλίδας (bubble sort) είναι ένας από τους πιο αποδοτικούς αλγόριθμους ταξινόμησης.	
στ)	Ο αλγόριθμος ταξινόμησης με εισαγωγή (insertion sort) είναι πιο αργός από τον αλγόριθμο ταξινόμησης φυσαλίδας (bubble sort).	
ζ)	Οι αλγόριθμοι ταξινόμησης φυσαλίδας (bubble sort) και εισαγωγής (insertion sort) δεν μπορούν να ταξινομήσουν σε φθίνουσα σειρά αριθμούς.	
η)	Ο αλγόριθμος σειριακής αναζήτησης (sequential search) μπορεί να χρησιμοποιηθεί μόνο για ακέραιους και πραγματικούς αριθμούς.	

2. Να χαρακτηρίσετε με σωστό (Σ) ή λάθος (Λ) καθεμιά από τις παρακάτω δηλώσεις μεταβλητών της γλώσσας προγραμματισμού C++.

A/A	Δήλωση μεταβλητών	Σωστό / Λάθος
(α)	<code>string names[3] = {"George", "Antonis"};</code>	
(β)	<code>double points[3] = {20.5, 13.2, 45.3, 35.3};</code>	
(γ)	<code>int points[3] [2]= {20, 13, 45};</code>	
(δ)	<code>char characters[3] [2] = { };</code>	

3. Σας δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++:

```
#include <iostream>
using namespace std;
int main(){
int i,j,pl=0,sum=0;


A


for (i=0;i<3;i++)
for (j=0;j<4;j++)
{
cout<<"Δώσε Αριθμό:";


B



Γ


}
for (i=0;i<3;i++)
for (j=0;j<4;j++)
{


Δ


pl++;
}
cout<<"Πλήθος θετικών τριψήφιων αριθμών:"<<pl<<endl;
cout<<"Άθροισμα όλων των στοιχείων του πίνακα:"<<sum<<endl;
return 0;
}
```

Να γράψετε:

(α) Την εντολή που πρέπει να τοποθετηθεί στη θέση A για να δηλωθεί ο δισδιάστατος πίνακας ακεραίων αριθμών, 3 γραμμών και 4 στηλών με το όνομα num.

.....

(β) Την εντολή που πρέπει να τοποθετηθεί στη θέση B με την οποία να καταχωρίζει στον πίνακα num, τους δώδεκα (12) ακέραιους αριθμούς τους οποίους θα δίνει ο χρήστης.

.....

(γ) Την εντολή που πρέπει να τοποθετηθεί στη θέση Γ, έτσι ώστε να υπολογίζεται το άθροισμα όλων των στοιχείων του πίνακα num.

.....

(δ) Τη συνθήκη που πρέπει να τοποθετηθεί στη θέση Δ, έτσι ώστε να υπολογίζεται το πλήθος των θετικών τριψήφιων αριθμών που βρίσκονται στον πίνακα num.

.....

4. Δίνεται ο πίνακας $\text{int arr}[3][3] = \{ \{0, 3, 6\}, \{5, 1\}, \{7, 2, -4\} \}$. Ποια θα είναι τα στοιχεία του πίνακα $\text{results}[3][3] = \{ 0 \}$ μετά την εκτέλεση των πιο κάτω εντολών της C++.

α) $\text{results}[2][0] = \text{arr}[2][2] + \text{arr}[1][2] * \text{arr}[2][2]$ (1,5 Mov.)

.....

.....

β) $\text{results}[2][1] = \text{arr}[0][2] \% \text{arr}[1][0] + \text{arr}[0][0]$ (1,5 Mov.)

.....

.....

γ) $\text{results}[1][1] = \text{arr}[2][0] + \text{arr}[0][1]$ (1,5 Mov.)

.....

.....

δ) $\text{results}[0][2] = \text{arr}[1][0] + \text{arr}[2][2] / \text{arr}[2][1]$ (1,5 Mov.)

.....

.....

ε) Να συμπληρώσετε τον πιο κάτω πίνακα (results) με όλα τα στοιχεία του.

(2 Mov.)

	0	1	2
0			
1			
2			

Πίνακας results

5. Να απαντήσετε τις πιο κάτω ερωτήσεις που αφορούν τον αντικειμενοστραφή προγραμματισμό.

(α) Τι ονομάζουμε κλάση (class); (2 Μον.)

.....
.....
.....
.....
.....

(β) Τι ονομάζουμε αντικείμενο (object); (2 Μον.)

.....
.....
.....
.....
.....

(γ) Τι είναι οι ιδιότητες (properties) και οι μέθοδοι (methods) στα αντικείμενα; (4 Μον.)

.....
.....
.....
.....
.....
.....
.....
.....

6. Δίνεται η παρακάτω κλάση η οποία χρησιμοποιείται για τη διαχείριση των ATM διαφόρων τραπεζών:

```
class atm
{
private:
float balance;
bool extra;
public:
string trapeza;
atm( )
{
balance = 3000;
}
```

```

void analipsi(float money)
{
    if (money <= balance)
        balance = balance - money;
    else
        cout << " Το ποσό ανάληψης υπερβαίνει το υπόλοιπο του λογαριασμού!" << endl;
}
void katathesi(float money)
{
    balance += money;
}
float vpoloipo()
{
    return balance;
}
};

```

Να απαντήσετε στις πιο κάτω ερωτήσεις:

(α) Να αναφέρετε ένα ιδιωτικό μέλος της κλάσης atm: (1,5 Μον.)

.....

(β) Να γράψετε μια εντολή με την οποία δημιουργούνται δύο αντικείμενα της πιο πάνω κλάσης με ονόματα at1 και at2: (1,5 Μον.)

.....

(γ) Υποθέτουμε ότι υπάρχει ένα αντικείμενο της πιο πάνω κλάσης με όνομα at1. Να γράψετε μια εντολή με την οποία θα καταχωριστεί στη μεταβλητή-μέλος trapeza του αντικειμένου at1 το κείμενο «ΤΕΣΕΚbank»: (1,5 Μον.)

.....

(δ) Υποθέτουμε ότι υπάρχει ένα αντικείμενο της παραπάνω κλάσης με όνομα at1. Τι θα κάνει η εντολή at1.poso=1000; Αιτιολογήστε την απάντησή σας. (2 Μον.)

.....

.....

.....

(ε) Υποθέτουμε ότι υπάρχει ένα αντικείμενο της παραπάνω κλάσης με όνομα at1. Να γράψετε μία εντολή με την οποία θα εμφανίζεται στην οθόνη το υπόλοιπο χρημάτων που υπάρχει στο at1. (1,5 Μον.)

.....

ΜΕΡΟΣ Β΄: Αποτελείται από τέσσερις (4) ερωτήσεις. Κάθε ορθή απάντηση βαθμολογείται με δέκα (10) μονάδες.

7. Δίδεται το πιο κάτω πρόγραμμα, στη γλώσσα προγραμματισμού C++. Το πρόγραμμα καταχωρίζει τα ονόματα δεκαπέντε (15) επιλεγόμενων μαθημάτων σε ένα μονοδιάστατο πίνακα με το όνομα `epilegomena` και το πλήθος των μαθητών που επέλεξαν το κάθε ένα από αυτά σε ένα άλλο παράλληλο μονοδιάστατο πίνακα με το όνομα `plith`. Ακολουθώς ταξινομεί τους δύο πίνακες σε αύξουσα σειρά με βάση το πλήθος, χρησιμοποιώντας τον αλγόριθμο εισαγωγής (`insertion sort`).

```
#include <iostream>
#define N 15
using namespace std;
int main ( ) {
```

A

```
for (i=0; i<N; i++)
    cin>>epilegomena[i]>>plith[i];
for (i=1;i<N;i++) {
    temp=epilegomena[i];
    tempa=plith[i];
    k=i-1;
    {
        plith[k+1]=plith[k];
        epilegomena[k+1]=epilegomena[k];
        k--;
    }
```

B

Γ

```
    }
for (i=0;i<N;i++)
    cout<<epilegomena[i]<<" "<<plith[i]<<endl;
return 0; }
```

Να γράψετε:

- (α) Την/τις εντολή/ές που πρέπει να τοποθετηθεί/ούν στη θέση A έτσι ώστε να δηλώνονται οι μεταβλητές του προγράμματος. (3 Μον.)

.....
.....

(β) Την/τις εντολή/ές που πρέπει να τοποθετηθεί/ούν στη θέση Β έτσι ώστε οι παράλληλοι πίνακες *epilegomena* και *plith* να ταξινομούνται σε αύξουσα σειρά (με βάση το πλήθος των μαθητών). (3 Μον.)

.....
.....

(γ) Την/τις εντολή/ές που πρέπει να τοποθετηθεί/ούν στη θέση Γ ώστε να υλοποιηθεί η ταξινόμηση των δύο παράλληλων πινάκων *epiligomena* και *plith*. (4 Μον.)

.....
.....

8. Να γράψετε συνάρτηση στην γλώσσα προγραμματισμού C++ που να επιστρέφει πόσες φορές εμφανίζεται ένας χαρακτήρας σε ένα πίνακα χαρακτήρων με την μέθοδο της σειριακής αναζήτησης (*sequential search*). Ο πίνακας, το μέγεθος του πίνακα και ο χαρακτήρας να δίνονται ως παράμετροι στην συνάρτηση.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

9. Πιο κάτω παρουσιάζονται δύο συναρτήσεις στη γλώσσα προγραμματισμού C++. Η πρώτη συνάρτηση εκτελεί ταξινόμηση μονοδιάστατου πίνακα πραγματικών αριθμών κατά αύξουσα σειρά χρησιμοποιώντας τον αλγόριθμο της φυσαλίδας (*Bubble Sort*) και η δεύτερη εκτελεί αναζήτηση ενός πραγματικού αριθμού και επιστρέφει την θέση του στον ταξινομημένο πίνακα χρησιμοποιώντας τον αλγόριθμο δυαδικής αναζήτησης (*Binary Search*). Αν δεν εντοπιστεί ο αριθμός επιστρέφει την τιμή -1. Στον πίνακα δεν υπάρχει αριθμός που να επαναλαμβάνεται. Στις συναρτήσεις υπάρχουν τέσσερα (4) λογικά λάθη. Να τα εντοπίσετε και να τα διορθώσετε συμπληρώνοντας στον πιο κάτω πίνακα τον αριθμό της γραμμής και την διορθωμένη εντολή. Στις συναρτήσεις να μη γίνει καμία προσθήκη ή αφαίρεση εντολής.


```

1 void bubbleSort(double arr[ ], int N) {
2     double temp;
3     for (int i=0; i<N-1; i++) {
4         for (int j=i+1; j<N; j++) {
5             if (arr[ i ]<arr[ j ]) {
6                 arr[ i ] = temp;
7                 arr[ i ] = arr[ j ];
8                 arr[ j ] = temp;
9             }
10        }
11    }
12 }
13 int binarySearch(double arr[ ], int left, int right, double key) {
14     while (left < right) {
15         int mid = right + left ;
16         if (arr[mid] == key)
17             return mid;
18         else if (key < arr[mid])
19             right = mid - 1;
20         else
21             left = mid + 1;
22     }
23     return -1;
24 }

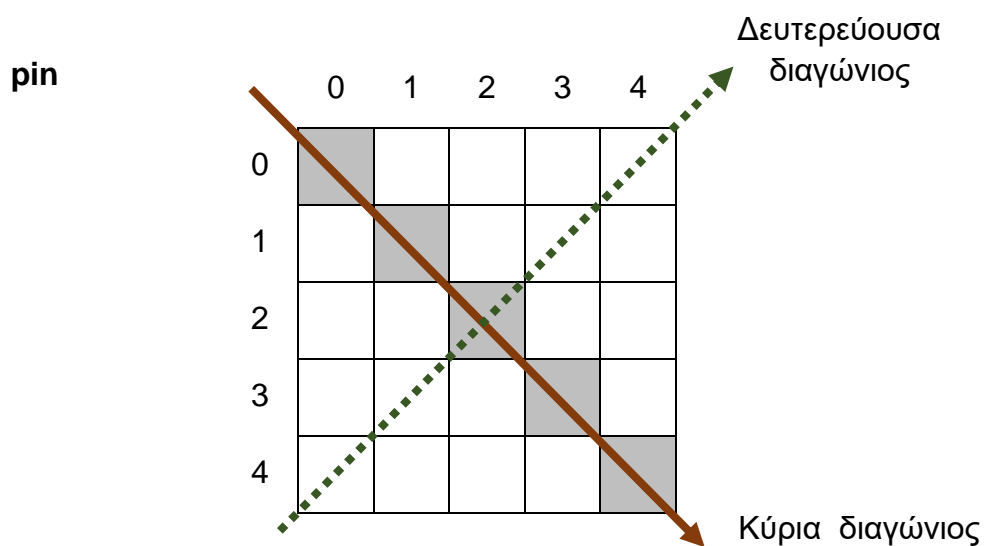
```

A/A	Αριθμός Γραμμής	Διορθωμένη εντολή
1		
2		
3		
4		

10. Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού C++ το οποίο:

(α) Να ζητά **25 ακέραιους** αριθμούς και να τους αποθηκεύει σε ένα τετραγωνικό πίνακα **5X5** με το όνομα **pin**. Οι αριθμοί να αποθηκεύονται κατά στήλη, δηλαδή να γεμίζει πρώτα η 1^η στήλη του πίνακα, μετά η 2^η στήλη κ.τ.λ.

(β) Να υπολογίζει και να επιστρέφει τον μεγαλύτερο αριθμό της κύριας διαγώνιου με τη βοήθεια της συνάρτησης **smax**. Η συνάρτηση **smax** θα δέχεται ως παράμετρο τον πίνακα **pin** από την κύρια συνάρτηση (κυρίως πρόγραμμα) και θα υπολογίζει και θα επιστρέφει στη κύρια συνάρτηση τον μεγαλύτερο αριθμό της κύριας διαγώνιου. Ο αριθμός αυτός θα τυπώνεται από τη κύρια συνάρτηση (κυρίως πρόγραμμα).



Το πρόγραμμα πρέπει να εμφανίζει στην οθόνη τα κατάλληλα μηνύματα για την εξαγωγή των αποτελεσμάτων σύμφωνα με το πιο κάτω παράδειγμα:

Παράδειγμα Εξόδου

(οθόνη)

Δώσε 25 ακέραιους αριθμούς:

Μέγιστος αριθμός κύριας διαγώνιου:

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

ΜΕΡΟΣ Γ΄: Αποτελείται από μία (1) ερώτηση. Η ορθή απάντηση βαθμολογείται με δώδεκα (12) μονάδες.

11. Να γράψετε πρόγραμμα σε γλώσσα προγραμματισμού C++ το οποίο :

- (α) Να δημιουργεί μια κατηγορία (κλάση), με όνομα Student για μαθητές. (2 Μον.)
- (β) Στα ιδιωτικά πεδία της κλάσης Student, να περιλαμβάνονται τα ακόλουθα: (2 Μον.)
- Όνοματεπώνυμο μαθητή
 - Τμήμα μαθητή
 - Όνομα Μαθήματος
 - Ένας μονοδιάστατος πίνακας τριών (3) ακεραίων τιμών όπου θα καταχωρούνται οι βαθμοί που αφορούν τις βαθμολογίες του μαθητή στο πρώτο τετράμηνο, στο δεύτερο τετράμηνο και στην τελική εξέταση.
- (γ) Στα δημόσια πεδία της κλάσης Student, να περιλαμβάνονται τα ακόλουθα: (6 Μον.)
- Ένας κατασκευαστής (constructor), ο οποίος θα ενημερώνει τα πεδία της κλάσης με τιμές που θα του παρέχονται ως παράμετροι κατά τη δημιουργία του.
 - Μια μέθοδος με όνομα mesos() που θα υπολογίζει τον μέσο όρο των βαθμών του μαθητή και θα τον επιστρέφει στο κυρίως πρόγραμμα (συνάρτηση main).
 - Μια μέθοδος με όνομα details() που θα εμφανίζει όλα τα στοιχεία του μαθητή.
- (δ) Στη κύρια συνάρτηση να υλοποιείται ένα αντικείμενο της κλάσης Student με τα ακόλουθα χαρακτηριστικά: (1 Μον.)
- Όνοματεπώνυμο: Γιώργος Γεωργίου
Τμήμα: ΘΗΨ3
Όνομα Μαθήματος: Εφαρμογές Προγραμματισμού III
Βαθμοί: 15, 10, 20
- (ε) Η κύρια συνάρτηση με τη βοήθεια των μεθόδων της κλάσης Student να εμφανίζει όλα τα στοιχεία του μαθητή και τον μέσο όρο της βαθμολογίας του μαθητή με δύο δεκαδικά ψηφία. (1 Μον.)

Παράδειγμα εξόδου

Όνοματεπώνυμο: Γιώργος Γεωργίου
Τμήμα: ΘΗΨ3
Όνομα Μαθήματος: Εφαρμογές Προγραμματισμού III
Μέσος όρος Βαθμολογίας: 15.00

.....
.....
.....

Dotted lines for writing.

