

ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ, ΠΟΛΙΤΙΣΜΟΥ, ΑΘΛΗΤΙΣΜΟΥ ΚΑΙ ΝΕΟΛΑΙΑΣ
ΔΙΕΥΘΥΝΣΗ ΜΕΣΗΣ ΓΕΝΙΚΗΣ ΕΚΠΑΙΔΕΥΣΗΣ

ΔΕΙΓΜΑ
ΕΝΙΑΙΑΣ ΓΡΑΠΤΗΣ ΑΞΙΟΛΟΓΗΣΗΣ Β' ΤΕΤΡΑΜΗΝΟΥ 2021-22
Γ' ΤΑΞΗΣ ΛΥΚΕΙΟΥ

ΛΥΣΕΙΣ

ΗΜΕΡΑ, __ ΜΑΪΟΥ 2022

ΕΞΕΤΑΖΟΜΕΝΟ ΜΑΘΗΜΑ:
ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΕΠΙΣΤΗΜΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ (Α' ΣΕΙΡΑ)

ΚΩΔΙΚΟΣ ΜΑΘΗΜΑΤΟΣ: Γ015

ΣΥΝΟΛΙΚΗ ΔΙΑΡΚΕΙΑ ΓΡΑΠΤΗΣ ΕΞΕΤΑΣΗΣ: 90 ΛΕΠΤΑ

ΤΟ ΕΞΕΤΑΣΤΙΚΟ ΔΟΚΙΜΙΟ ΑΠΟΤΕΛΕΙΤΑΙ ΑΠΟ ΕΠΤΑ (7) ΣΕΛΙΔΕΣ

ΟΔΗΓΙΕΣ (για τους εξεταζομένους)

1. Στο εξώφυλλο του τετραδίου να συμπληρώσετε **όλα τα κενά** με τα στοιχεία που ζητούνται.
2. Το εξεταστικό δοκίμιο αποτελείται από **δύο (2)** μέρη **Α'** και **Β'**.
3. Το **ΜΕΡΟΣ Α'** αποτελείται από **τέσσερις (4)** ασκήσεις. Να απαντήσετε **ΟΛΕΣ** τις ασκήσεις. Κάθε άσκηση βαθμολογείται με **δεκαπέντε (15)** μονάδες.
4. Το **ΜΕΡΟΣ Β'** αποτελείται από **δύο (2)** ασκήσεις. Να απαντήσετε **ΟΛΕΣ** τις ασκήσεις. Κάθε άσκηση βαθμολογείται με **είκοσι (20)** μονάδες.
5. Να **μη αντιγράψετε τα θέματα** στο τετράδιο απαντήσεων.
6. Να **μη γράψετε πουθενά** στις απαντήσεις σας το **όνομά** σας.
7. Να απαντήσετε στο **τετράδιό** σας σε όλα τα θέματα **μόνο με μπλε πένα ανεξίτηλης μελάνης**. Τα σύμβολα των **Λογικών Κυκλωμάτων**, καθώς και το **λεκτικό περιεχόμενό** τους μπορούν να γίνουν με **μολύβι**.
8. **Επιτρέπεται** η χρήση **μη προγραμματιζόμενης υπολογιστικής μηχανής** που φέρει τη **σφραγίδα** του σχολείου.
9. **Απαγορεύεται** η χρήση **διορθωτικού υγρού** ή **διορθωτικής ταινίας**.
10. Οι μοναδικές βιβλιοθήκες που επιτρέπονται στη δημιουργία προγραμμάτων, είναι η **<iostream>**, **<fstream>**, **<string>**, **<iomanip>**, **<cmath>** και **<climits>**.
11. Η έκδοση της γλώσσας C++ που μπορεί να χρησιμοποιηθεί είναι η **C++98 (ISO/IEC 14882:1998)**. Οποιοσδήποτε επεκτάσεις (extensions) παρέχονται από κάποιους μεταγλωττιστές (compilers) δεν μπορούν να χρησιμοποιηθούν.

ΜΕΡΟΣ Α': Αποτελείται από 4 ασκήσεις. Να λύσετε ΟΛΕΣ τις ασκήσεις.
Κάθε άσκηση βαθμολογείται με 15 μονάδες.

ΑΣΚΗΣΗ 1:

Δίνονται ο δεκαδικός αριθμός **A=93** και οι δυαδικοί αριθμοί **B=10110001** και **Γ=01110100**.

(α) Να δείξετε ότι η αντίστοιχη τιμή του **δεκαδικού αριθμού A** στο δυαδικό σύστημα είναι **(1011101)₂**, **σημειώνοντας τα βήματα** που ακολουθήσατε για να φτάσετε στο συγκεκριμένο αποτέλεσμα.

(Μονάδες 5)

(β) Αφού υπολογίσετε το **συμπλήρωμα ως προς 2** του δυαδικού αριθμού **Γ** να γράψετε στο δυαδικό σύστημα το αποτέλεσμα της **αφαίρεσης B-Γ**.

(Μονάδες 5)

(γ) Να δείξετε ότι η αντίστοιχη τιμή του **δυαδικού αριθμού B** στο δεκαδικό σύστημα είναι **(177)₁₀**, **σημειώνοντας τα βήματα** που ακολουθήσατε για να φτάσετε στο συγκεκριμένο αποτέλεσμα.

(Μονάδες 5)

ΛΥΣΗ:

(α) Μετατροπή του δεκαδικού αριθμού **A=93** στο δυαδικό σύστημα.

	Αριθμός	Πηλίκο	Υπόλοιπο	
Βήμα 1:	93/2	46	1	↑
Βήμα 2:	46/2	23	0	
Βήμα 3:	23/2	11	1	
Βήμα 4:	11/2	5	1	
Βήμα 5:	5/2	2	1	
Βήμα 6:	2/2	1	0	
Βήμα 7:	1/2	0	1	

$$(93)_{10} = (1011101)_2$$

(β) Πρόσθεση των δυαδικών αριθμών **B=10110001** και **Γ=01110100**

Συμπλήρωμα ως προς 2 του αριθμού Γ: 10001100

$$\begin{array}{r}
 1\ 0\ 1\ 1\ 0\ 0\ 0\ 1 \\
 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0 \\
 \hline
 4\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1
 \end{array}
 + \left. \begin{array}{l} \\ \\ \\ \\ \\ \\ \\ \end{array} \right\} B-\Gamma = (00111101)_2$$

(γ) Μετατροπή του δυαδικού αριθμού **B=(10110001)₂** στο δεκαδικό σύστημα.

2 ⁷	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰
128	64	32	16	8	4	2	1
1	0	1	1	0	0	0	1

$$\begin{aligned}
 B=(10110001)_2 &= 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\
 &= 128 + 32 + 16 + 1 \\
 &= (177)_{10}
 \end{aligned}$$

ΑΣΚΗΣΗ 2:

Δίνεται η πιο κάτω λογική συνάρτηση:

$$F(A,B,C) = A'BC' + A'B'C' + AB'C' + AB$$

(α) Να δημιουργήσετε τον **πίνακα αληθείας** της συνάρτησης.

(Μονάδες 4)

(β) Να σχεδιάσετε τον **χάρτη Karnaugh** που αντιστοιχεί στη συνάρτηση.

(Μονάδες 4)

(γ) Να **απλοποιήσετε** τη συνάρτηση (στην πιο απλή της μορφή), με τη χρήση **χάρτη Karnaugh** και να σχεδιάσετε το **λογικό κύκλωμα** που θα προκύψει μετά την απλοποίηση.

(Μονάδες 7)

ΛΥΣΗ:

$$F(A,B,C) = A'BC' + A'B'C' + AB'C' + AB$$

(α) **Πίνακας αληθείας** της συνάρτησης F.

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

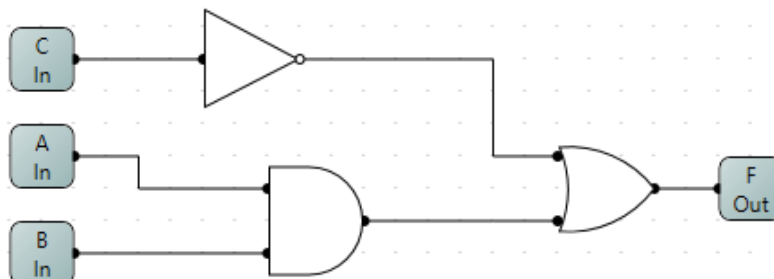
(β) **Πίνακας Karnaugh** που αντιστοιχεί στη συνάρτηση F.

A \ BC	00	01	11	10
0	1	0	0	1
1	1	0	1	1

(γ) **Απλοποίηση F** και αντίστοιχο **λογικό κύκλωμα**

A \ BC	00	01	11	10
0	1	0	0	1
1	1	0	1	1

$$F(A,B,C) = C' + AB$$



ΑΣΚΗΣΗ 3:

Ο Παναγιώτης στην προσπάθεια του να οργανώσει τη βιβλιοθήκη του, σας ζήτησε να του ετοιμάσετε ένα πρόγραμμα στο οποίο θα καταχωρήσει τους **τίτλους** των **500** βιβλίων που υπάρχουν σε αυτή και ακολούθως δίνοντας τον **τίτλο** ενός βιβλίου, να βρίσκει αν το βιβλίο αυτό υπάρχει στην βιβλιοθήκη ή όχι.

Να γράψετε πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο:

(α) Να **δέχεται** τους **τίτλους** των 500 βιβλίων που υπάρχουν στην βιβλιοθήκη (**ο κάθε τίτλος μπορεί να αποτελείται από περισσότερες από μια λέξη**) και να τους αποθηκεύει στον μονοδιάστατο πίνακα **books**.

(Μονάδες 3)

(β) Να **ταξινομεί** τον πίνακα **books** σε **αλφαβητική** σειρά με τη χρήση του αλγόριθμου **Εισαγωγής (Insertion Sort)**.

(Μονάδες 6)

(γ) Να **δέχεται** τον **τίτλο** ενός βιβλίου και με τη χρήση του αλγόριθμου **Διαδικής Αναζήτησης (Binary Search)** να γίνεται αναζήτηση για εντοπισμό του συγκεκριμένου βιβλίου μέσα στον πίνακα **Books**. Μόλις εντοπιστεί το βιβλίο, ο αλγόριθμος διαδικής αναζήτησης θα πρέπει να **σταματά** και να εμφανίζεται το μήνυμα: **«Υπάρχει μέσα στη βιβλιοθήκη»**. Στην περίπτωση που **δεν εντοπιστεί το βιβλίο**, να εμφανίζεται το μήνυμα: **«Δεν υπάρχει μέσα στη βιβλιοθήκη»**.

(Μονάδες 6)

ΛΥΣΗ:

```
#include <iostream>
#include <string>
#define N 500
using namespace std;
int main() {
    int i, k, first=0, last=N-1, mid;
    string temp, books[N], bk;
    bool flag=false;

    cout<<"Δώσε τους τίτλους των βιβλίων: ";
    for (i=0; i<N; i++)
        getline(cin, books[i]);

    for (i=1; i<N; i++) {
        temp=books[i];
        k=i-1;
        while (k>=0 && books[k]>temp) { //αύξουσα σειρά (Αλφαβητική σειρά)
            books[k+1]=books[k];
            k--;
        }
        books[k+1]= temp;
    }
    cout<<"Δώσε τον τίτλο του βιβλίου που ψάχνεις:";
    getline(cin, bk);

    while (first<=last && flag==false) {
```

```

    mid=(first+last)/2;
    if (bk==books[mid])
        flag=true;
    else if (bk<books[mid])
        last=mid-1;
    else
        last=mid+1;
}

if (flag==true)
    cout<<"Υπάρχει μέσα στη βιβλιοθήκη";
else
    cout<<"Δεν υπάρχει μέσα στη βιβλιοθήκη ";
return 0;
}

```

ΑΣΚΗΣΗ 4:

Σας δίνεται ο πιο κάτω **δισδιάστατος** πίνακας **3 γραμμών** και **3 στηλών** με το όνομα **num** και το πιο κάτω πρόγραμμα στην γλώσσα προγραμματισμού C++.

		0	1	2
num	0	31	11	177
	1	22	14	61
	2	27	362	83

```

#include <iostream>
using namespace std;
int main(){


A


    int min=num[1][2], plithos=0;
    for (int i=0;i<3;i++)
        for (int j=0;j<3;j++){


B


            min=num[i][j];


Γ


            plithos++;
        }
    cout<<"Μικρότερος αριθμός κάτω από δευτερεύουσα
    διαγώνιο:"<<min<<endl;
    cout<<"Πλήθος διψήφιων αριθμών κύριας διαγωνίου :"<<plithos;
    return 0;
}

```

Να γράψετε στο τετράδιο απαντήσεών σας:

(α) Την **εντολή** που πρέπει να τοποθετηθεί στη **θέση Α** για να **δηλωθεί** ο πίνακας **num**, αρχικοποιώντας τον με τις **τιμές** που περιέχονται σ' αυτόν.

(Μονάδες 3)

(β) Την **εντολή** που πρέπει να τοποθετηθεί στη **θέση Β**, έτσι ώστε να **υπολογίζεται** και να **τυπώνεται** ο **μικρότερος αριθμός** που βρίσκεται **κάτω** από την **δευτερεύουσα διαγώνιο**.

(Μονάδες 6)

(γ) Την **εντολή** που πρέπει να τοποθετηθεί στη **θέση Γ**, έτσι ώστε να **υπολογίζεται** το **πλήθος** των **διψήφιων** αριθμών που βρίσκονται **στη κύρια διαγώνιο**.

(Μονάδες 6)

ΛΥΣΗ:

(α) **Θέση Α:** `int num[3][3]={{32,12,7},{11,14,41},{26,67,72}};`

(β) **Θέση Β:** `if (i<j && num[i][j]>max)`

(γ) **Θέση Γ:** `if (i+j==2 && num[i][j]%2==0)`

**ΤΕΛΟΣ Α' ΜΕΡΟΥΣ
ΑΚΟΛΟΥΘΕΙ ΤΟ ΜΕΡΟΣ Β'**

ΜΕΡΟΣ Β': Αποτελείται από 2 ασκήσεις. Να λύσετε ΟΛΕΣ τις ασκήσεις.
Κάθε άσκηση βαθμολογείται με 20 μονάδες.

ΑΣΚΗΣΗ 5:

Ο Ευρωπαϊκός διαγωνισμός ζωγραφικής EuroPaintArt είναι μια πρωτοβουλία που ξεκίνησε στην Ευρώπη το 2015. Στον διαγωνισμό, χρόνο με τον χρόνο, συμμετέχουν ολοένα και περισσότερες χώρες, με αποκορύφωμα τον Ιανουάριο του 2020 κατά τον οποίο συμμετείχαν **35** χώρες, οι περισσότερες συμμετοχές μέχρι σήμερα. Τα ονόματα των 35 αυτών χωρών αποθηκεύονται στον μονοδιάστατο πίνακα **countries** και οι βαθμολογίες που είχαν πάρει σε κάθε χρονιά σ' έναν άλλο πίνακα **ακέραιων** αριθμών **δύο διαστάσεων**, ο οποίος έχει το όνομα **results** και είναι παράλληλος με τον πρώτο πίνακα. Στην περίπτωση που μια χώρα δεν συμμετείχε μια χρονιά στον διαγωνισμό, τότε στην αντίστοιχη θέση του πίνακα **results** καταχωρείται ο **αριθμός 0**. Επιπλέον, οι βαθμολογίες που λαμβάνουν για κάθε χρονιά οι χώρες που συμμετέχουν στον διαγωνισμό, είναι **μοναδικές**, δηλαδή δεν υπάρχει χώρα που να πήρε την ίδια βαθμολογία με μια άλλη την **ίδια χρονιά**.

Παράδειγμα:

countries		results						
		0	1	2	3	4	5	
0	Greece	0	0	85	67	72	90	
1	Spain	1	78	92	88	96	80	87
2	France	2	80	0	0	92	93	79
.	
.	
33	Cyprus	33	0	86	0	65	0	88
34	Italy	34	90	88	91	88	81	94

← Η 1^η γραμμή (γραμμή 0) του πίνακα **results** παρουσιάζει τις βαθμολογίες που πήρε η Ελλάδα στον διαγωνισμό. Σύμφωνα με το παράδειγμα η Ελλάδα **δεν συμμετείχε** στον διαγωνισμό το **2015** (στήλη 0) και το **2016** (στήλη 1) και άρα για τις χρονιές αυτές έχει καταχωριστεί ο αριθμός 0.

Να γράψετε πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο:

- (α) Να **καταχωρίζει** στον μονοδιάστατο πίνακα **countries** τα **ονόματα** των **χωρών** και στον παράλληλο διδιάστατο πίνακα **results** τις **βαθμολογίες** που πήρε η κάθε χώρα από το **2015** μέχρι και το **2020**. Να θεωρήσετε ότι τα δεδομένα δίνονται σωστά και δεν χρειάζεται οποιοσδήποτε έλεγχος. **(Μονάδες 4)**
- (β) Να **υπολογίζει** τον **μέσο όρο** βαθμολογίας που πήρε όλα αυτά τα χρόνια κάθε χώρα και να τον **καταχωρίζει** σ' έναν άλλο παράλληλο μονοδιάστατο πίνακα με το όνομα **average**. Στον υπολογισμό του μέσου όρου να **μη** λαμβάνονται υπόψη οι χρονιές που μια χώρα **δεν** έχει λάβει μέρος. Δηλαδή, σύμφωνα και με το πιο πάνω παράδειγμα, ο μέσος όρος για την Ελλάδα, θα υπολογιστεί λαμβάνοντας υπόψη πως η συγκεκριμένη χώρα συμμετείχε στον διαγωνισμό μόνο **4** χρονιές. **(Μονάδες 8)**
- (γ) Να **καταχωρίζει** στον μονοδιάστατο πίνακα **minCountry** το **όνομα της χώρας** που έχει πάρει τη **μικρότερη βαθμολογία** για κάθε χρονιά από το 2015 μέχρι και το 2020. Στη συνέχεια να **εκτυπώνει** τα περιεχόμενα του πίνακα. Σύμφωνα με το πιο πάνω παράδειγμα και λαμβάνοντας υπόψη μόνο τις πέντε χώρες που αναγράφονται, το **2015** (στήλη 0) η χώρα με τη μικρότερη βαθμολογία είναι η **Ισπανία (Spain)**, το **2016** (στήλη 1) η **Κύπρος (Cyprus)** κ.ο.κ. **(Μονάδες 8)**

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

Παράδειγμα Εισόδου	(για 5 χώρες μόνο)
Greece 0 0 85 67 72 90	
Spain 78 92 88 96 80 87	
France 80 0 0 92 93 79	
Cyprus 0 86 0 65 0 88	
Italy 90 88 91 88 81 94	
Παράδειγμα Εξόδου (οθόνη)	(για 5 χώρες μόνο)
Δώσε όνομα και βαθμολογίες κατά έτος κάθε χώρας:	
Οι χώρες με τη μικρότερη βαθμολογία κατά έτος:	
Spain Cyprus Greece Cyprus Greece France	

ΛΥΣΗ:

```
#include<iostream>
#include<iomanip>
#include<climits>
using namespace std;
#define row 35
#define col 6

int main(){
    string countries[row],minCountry[col],minname,tempa;
    int i,j,results[row][col],sum,c,min;
    float average[row],temp;
    bool sorted;

    cout<<"Δώσε όνομα και βαθμολογίες κατά έτος κάθε χώρας:"<<endl;
    for (i=0;i<row;i++){
        cin>>countries[i];
        for (j=0;j<col;j++)
            cin>>results[i][j];
    }
    for (i=0;i<row;i++){
        sum=0;c=0;
        for (j=0;j<col;j++)
            if (results[i][j]!=0){
                sum+=results[i][j];
                c++;
            }
        average[i]=(float)sum/c;
    }
    for (j=0;j<col;j++){
        min=INT_MAX;
        for (i=0;i<row;i++)
            if (results[i][j]<min && results[i][j]!=0){
                min=results[i][j];
                minname=countries[i];
            }
        minCountry[j]=minname;
    }
```



```

}
cout<<"Οι χώρες με τη μικρότερη βαθμολογία κατά έτος:"<<endl;
for (i=0;i<col;i++)
    cout<<minCountry[i]<<" ";
return 0;
}

```

ΑΣΚΗΣΗ 6:

Η εταιρεία παρασκευής κρασιού «Wein» διαθέτει μια μεγάλη κάβα με κρασιά που παρασκευάζει. Για την καλύτερη οργάνωση και διαχείριση του κύκλου εργασιών της, η εταιρεία σας ζητά να δημιουργήσετε ένα πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο:

(α) Να δημιουργεί μια εγγραφή με το όνομα **cava**, η οποία να περιλαμβάνει τα πιο κάτω μέλη:

- Κωδικός κρασιού (integer)
- Όνομα κρασιού (string)
- Έτος παραγωγής (integer)
- Διαθέσιμη ποσότητα σε μπουκάλια (integer)
- Τιμή πώλησης κρασιού ανά μπουκάλι (float)

Στη συνέχεια να διαβάζει από το αρχείο **askisi12IN.txt** τα στοιχεία (κωδικός κρασιού, όνομα κρασιού, έτος παραγωγής, διαθέσιμη ποσότητα, τιμή πώλησης) των **50** ειδών κρασιού που παρασκευάζει η εταιρεία και να τα **καταχωρίζει** σ' έναν πίνακα εγγραφών τύπου **cava**, με το όνομα **wine**.

(Μονάδες 6)

(β) Να **τυπώνει** σε **ξεχωριστή γραμμή** τα στοιχεία (κωδικός κρασιού, όνομα κρασιού, έτος παραγωγής, διαθέσιμη ποσότητα σε μπουκάλια, τιμή πώλησης ανά μπουκάλι) για κάθε είδος κρασιού του οποίου το έτος παραγωγής είναι **πριν το 2010** με τιμή πώλησής μεγαλύτερη από **100 ευρώ** και με **διαθέσιμη ποσότητα**. Αν δεν υπάρχουν τέτοια διαθέσιμα κρασιά, να εμφανίζει το μήνυμα «**Δεν υπάρχουν διαθέσιμα τέτοια κρασιά!**».

(Μονάδες 7)

(γ) Να **υπολογίζει** και να **τυπώνει** την **τιμή πώλησης** (με 2 δεκαδικά ψηφία), το **όνομα** και το **έτος παραγωγής** του πιο ακριβού κρασιού. Να θεωρήσετε ότι ένα είναι το πιο ακριβό κρασί.

(Μονάδες 7)

Το πρόγραμμα πρέπει να εμφανίζει στην οθόνη τα κατάλληλα μηνύματα για την εισαγωγή των δεδομένων και την εξαγωγή των αποτελεσμάτων σύμφωνα με τα πιο κάτω παραδείγματα:

Παράδειγμα Εισόδου 1 (από αρχείο askisi12IN.txt) 1 Aphrodita 2005 250 50.00 2 Inos 2009 0 150.00 3 Lemona 2015 140 20.50 4 Argos 2000 25 180.75 5 Linos 2019 1200 14.50	(για 5 είδη κρασιών μόνο)
Παράδειγμα Εξόδου 1 (οθόνη) Κρασιά παραγωγής πριν το 2010 με τιμή πώλησης πάνω από 100 ευρώ: 4 Argos 2000 25 180.75 Τιμή, όνομα, έτος παραγωγής πιο ακριβού κρασιού:180.75 Argos 2000	(για 5 είδη κρασιών μόνο)
Παράδειγμα Εισόδου 2 (από αρχείο askisi12IN.txt) 1 Aios 2016 650 35.00 2 Eirina 2010 63 118.00 3 Enos 2003 0 120.50 4 Orgos 2018 0 5.75 5 Lena 2000 100 99.50	(για 5 είδη κρασιών μόνο)
Παράδειγμα Εξόδου 2 (οθόνη) Κρασιά παραγωγής πριν το 2010 με τιμή πώλησης πάνω από 100 ευρώ: Δεν υπάρχουν διαθέσιμα τέτοια κρασιά! Τιμή, όνομα, έτος παραγωγής πιο ακριβού κρασιού:120.50 Enos 2003	(για 5 είδη κρασιών μόνο)

ΛΥΣΗ:

```
#include<iostream>
#include<iomanip>
#include<fstream>

using namespace std;

#define n 50

struct cava{
    int code;
    string name_k;
    int etos, posotita;
    float timi;
};

int main(){

    ifstream fin("askisi12IN.txt");
    ofstream fout("askisi12OUT.txt");
    int i,maxetos,tot_f=0;
    float maxtimi;
    string maxname_k;
    bool found=false;
    cava wine[n];
```

```

for(i=0;i<n;i++)
    fin>>wine[i].code>>wine[i].name_k>>wine[i].etos>>
        wine[i].posotita>>wine[i].timi;

cout<<"Κρασιά παραγωγής πριν το 2010 με τιμή πώλησης πάνω από 100
    Ευρώ:"<<endl;
for(i=0;i<n;i++)
    if(wine[i].etos<2010 && wine[i].timi>100 &&
wine[i].posotita>0){
        cout<<wine[i].code<<" "<<wine[i].name_k<<" "<<
            wine[i].etos<<" "<< wine[i].posotita<<" "<<
            fixed<<setprecision(2)<<wine[i].timi<<endl;
        found=true;
    }
if(found==false)
    cout<<"Δεν υπάρχουν τέτοια κρασιά!"<<endl;
maxtimi=0;
for(i=0;i<n;i++)
    if (wine[i].timi>maxtimi) {
        maxtimi=wine[i].timi;
        maxname_k=wine[i].name_k;
        maxetos=wine[i].etos;
    }
cout<<"Τιμή, όνομα, έτος παραγωγής πιο ακριβού
κρασιού:"<<fixed<<setprecision(2)<<maxtimi<<" "<<maxname_k<<" "
<<maxetos<<endl;

fin.close();
fout.close();
return 0;
}

```

**ΤΕΛΟΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ
ΣΑΣ ΕΥΧΟΜΑΣΤΕ ΚΑΛΗ ΕΠΙΤΥΧΙΑ**

**ΤΥΠΟΛΟΓΙΟ ΣΥΝΑΡΤΗΣΕΩΝ ΚΑΙ ΣΤΑΘΕΡΩΝ ΣΤΗ ΓΛΩΣΣΑ
ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ C++**

ΣΥΝΑΡΤΗΣΕΙΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ <cmath>		
Συνάρτηση	Χρήση	Παράμετροι
sqrt(x)	Επιστρέφει την τετραγωνική ρίζα του αριθμού x. Η επιστρεφόμενη τιμή είναι πραγματικός αριθμός.	Ένας θετικός αριθμός (ακέραιος ή πραγματικός)
abs(x)	Επιστρέφει την απόλυτη τιμή του αριθμού x. Η επιστρεφόμενη τιμή εξαρτάται από τον τύπο του αριθμού x.	Ένας αριθμός (ακέραιος ή πραγματικός)
pow(x,y)	Επιστρέφει το αποτέλεσμα της δύναμης x^y . Η επιστρεφόμενη τιμή είναι πραγματικός αριθμός.	Δύο πραγματικοί αριθμοί
trunc(x)	Επιστρέφει το ακέραιο μέρος του αριθμού x σε πραγματική μορφή, αγνοώντας το δεκαδικό μέρος του .	Ένας πραγματικός αριθμός
round(x)	Επιστρέφει το ακέραιο μέρος του αριθμού x σε πραγματική μορφή, στρογγυλοποιημένο στην πλησιέστερη τιμή .	Ένας πραγματικός αριθμός
ΣΥΝΑΡΤΗΣΕΙΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ <string>		
size()	Επιστρέφει το μέγεθος μιας συμβολοσειράς. Η επιστρεφόμενη τιμή είναι ακέραιος αριθμός που συμβολίζει από πόσα bytes αποτελείται μια συμβολοσειρά.	Καμία παράμετρος
clear()	Διαγράφει το περιεχόμενο μιας συμβολοσειράς. Δεν επιστρέφει τίποτα.	Καμία παράμετρος
empty()	Ελέγχει αν μια συμβολοσειρά είναι άδεια . Η επιστρεφόμενη τιμή είναι τύπου Boolean .	Καμία παράμετρος
getline(x,y)	Αποθηκεύει ολόκληρη μια συμβολοσειρά που μπορεί να εισαχθεί από το πληκτρολόγιο ή από αρχείο (x) στο αντικείμενο y.	Η 1 ^η παράμετρος (x) αφορά την μέθοδο εισαγωγής της συμβολοσειράς (π.χ. από το πληκτρολόγιο ή από αρχείο) και η 2 ^η παράμετρος (y) αφορά το αντικείμενο στο οποίο θα αποθηκευτεί η συμβολοσειρά η οποία έχει διαβαστεί αρχικά.
ΣΤΑΘΕΡΕΣ ΤΗΣ ΒΙΒΛΙΟΘΗΚΗΣ <climits>		
INT_MAX	Μέγιστο αριθμητικό όριο μεταβλητής ή σταθεράς τύπου integer . Η ακριβής αριθμητική τιμή της είναι 32767 (στα 2 bytes) ή 2147483647 (στα 4 bytes)	
INT_MIN	Ελάχιστο αριθμητικό όριο μεταβλητής ή σταθεράς τύπου integer . Η ακριβής αριθμητική τιμή της είναι -32767 (στα 2 bytes) ή 2147483647 (στα 4 bytes)	