

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

**ΕΝΙΑΙΕΣ ΤΕΛΙΚΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΚΑΙ
ΑΠΟΛΥΤΗΡΙΕΣ ΓΡΑΠΤΕΣ ΕΞΕΤΑΣΕΙΣ**

20 25 - 20 26

Β' ΤΑΞΗΣ ΤΕΣΕΚ

ΣΕΙΡΑ Α'

ΘΕΩΡΗΤΙΚΗ ΚΑΤΕΥΘΥΝΣΗ

ΗΜΕΡΟΜΗΝΙΑ : Πέμπτη, 21 Μαΐου 2026

ΕΞΕΤΑΖΟΜΕΝΟ ΜΑΘΗΜΑ: Εφαρμογές Προγραμματισμού II -TEM2

ΚΩΔΙΚΟΣ ΜΑΘΗΜΑΤΟΣ : thips202

ΛΥΣΕΙΣ

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

1. Οι μεταβλητές a και b είναι τύπου integer και έχουν τις ακόλουθες τιμές:

$$a = 3 \text{ και } b = -5$$

Να γράψετε την τιμή που θα έχει η λογική μεταβλητή x (boolean), όταν εκτελεστεί η πιο κάτω εντολή στη γλώσσα προγραμματισμού C++, σημειώνοντας τα βήματα που ακολουθήσατε για να φτάσετε στο συγκεκριμένο αποτέλεσμα.

$$x = \text{abs}(b+a) > 0 \ || \ \text{trunc}(a + 1.6) == 5$$

x= abs(-5+3) > 0 trunc(3 + 1.6) == 5	1 μονάδα
x= abs(-2) > 0 trunc(4.6) == 5	1 μονάδα
x= 2 > 0 4 == 5	2 μονάδες
x= true false	2 μονάδες
x= true	2 μονάδες

2. Δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++. Να παρουσιάσετε τι θα εμφανιστεί στην οθόνη μετά την εκτέλεσή του.

```
#include <iostream>
using namespace std;
int main( )
{
    for(int i=0; i<4; i++)
    {
        for(int j=0; j<=i; j++)
            cout<<"*";
        cout<<endl;
    }
    return 0;
}
*
**
***
****
```

Κάθε ορθή γραμμή (έξοδος) 2 μονάδες

3. Δίνονται οι πιο κάτω πίνακες:

Πίνακας A

10	20	30	40
----	----	----	----

Πίνακας B

0	0	0	0
---	---	---	---

(α) Να γράψετε τις εντολές για να δηλωθούν οι πίνακες A και B αρχικοποιώντας τους με τις τιμές που περιέχονται σε αυτούς.

int A[4] = {10, 20, 30, 40}; **2 μονάδες**

int B[4] = {0}; ή int B[4] = { }; ή int B[4] = { 0, 0, 0, 0}; **2 μονάδες**

(β) Να βρείτε τα αποτελέσματα των πιο κάτω εντολών και να συμπληρώσετε τον πιο κάτω πίνακα (B) με όλα τα νέα στοιχεία του.

- i. B[0] = A[1]; **20**
- ii. B[1] = A[0] + A[1]; **10 + 20 = 30**
- iii. B[2] = A[3] / A[1]; **40/20 = 2**
- iv. B[3] = A[3] % A[2]; **40 % 30 = 10**

Πίνακας B

20	30	2	10
-----------	-----------	----------	-----------

1 μονάδα η κάθε ορθή απάντηση σύνολο 4 μονάδες.

4. Δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++:

```
#include <fstream>
using namespace std;
int main( ) {
    ofstream fout1("data.txt");
    fout1 << "Θεωρητική"<<endl;
    fout1.close( );
    ofstream fout2("data.txt",(ios::out | ios::app));
    fout2 << "Κατεύθυνση"<<endl;
    fout2.close( );
    ofstream fout3("data.txt");
    fout3 << "Ψηφιακή Τεχνολογία"<<endl;
    fout3.close( );
    ofstream fout4 ("data.txt",(ios::out | ios::app));
    fout4 << "και Προγραμματισμός"<<endl;
    fout4.close( );
```

```
return 0; }
```

Να γράψετε το τελικό περιεχόμενο του αρχείου data.txt μετά την εκτέλεση του προγράμματος.

Ψηφιακή Τεχνολογία και Προγραμματισμός

Κάθε ορθή γραμμή (έξοδος) 4 μονάδες

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

A/A	Πρόταση	Σωστό / Λάθος
1.	Μια συνάρτηση πρέπει πάντα να επιστρέφει τιμή.	Λ
2.	Μια παράμετρος στο σημείο δήλωσης μιας συνάρτησης με την αντίστοιχη παράμετρο στο σημείο κλήσης της συνάρτησης πρέπει να είναι ίδιου τύπου.	Σ
3.	Οι συναρτήσεις διευκολύνουν τη συντήρηση και διόρθωση λαθών.	Σ
4.	Μια συνάρτηση δεν μπορεί να επιστρέφει περισσότερες από μία (1) τιμές στο σημείο κλήσης της.	Λ

2 μονάδες η κάθε ορθή απάντηση σύνολο 8 μονάδες.

6. Δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++.

```
#include <iostream>
using namespace std;

int a = 2;

void calc(int x, int &y)
{
    int z=1;
    x = x + 5;
    y = y + 5;
    a = a + x + z;
    cout << "x = " << x << endl;
}
```

```

int main( )
{
    int b = 3;
    int c = 5;
    calc(b, c);
    cout << "a = " << a << endl;
    cout << "b = " << b << endl;
    cout << "c = " << c << endl;
    return 0;
}

```

(α) Να εντοπίσετε όλες τις μεταβλητές και τις παραμέτρους του προγράμματος και να καταχωρίσετε τα ονόματά τους στον Πίνακα 1, κατηγοριοποιώντας τα στη σωστή στήλη.

Καθολικές Μεταβλητές	Τοπικές μεταβλητές κύριας συνάρτησης (main)	Συνάρτηση calc		
		Τοπικές μεταβλητές	Τυπικές παραμέτροι τιμής (by value)	Τυπικές παράμετροι αναφοράς (by reference)
a	b, c	z	x	y

Πίνακας 1

a → 0.5 μονάδα,

b, c → 1 (0.5 + 0.5) μονάδα

z → 0.5 μονάδα

x → 1 μονάδα

y → 1 μονάδα

Σύνολο 4 μονάδες

(β) Να γράψετε τι θα εμφανιστεί στην οθόνη μετά την εκτέλεση του πιο πάνω προγράμματος. Να δοθεί μόνο η τελική μορφή της εξόδου.

x = 8

a = 11

b = 3

c = 10

Κάθε ορθή γραμμή (έξοδος) 1 μονάδα (σύνολο 4 μονάδες).

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

7. Δίνεται το παρακάτω πρόγραμμα σε γλώσσα προγραμματισμού C++. Το πρόγραμμα διαχειρίζεται τα δεδομένα 100 μαθητών. Συγκεκριμένα, διαβάζει τα ονόματα και τους ακέραιους βαθμούς (τιμές από 1 έως 20) των μαθητών και τα αποθηκεύει σε δύο μονοδιάστατους πίνακες με ονόματα Names και Grades, αντίστοιχα. Στη συνέχεια, υπολογίζει τον μικρότερο βαθμό και εμφανίζει τα ονόματα όλων των μαθητών που έχουν τον βαθμό αυτό.

Στο πρόγραμμα υπάρχουν τέσσερα (4) συντακτικά ή/και λογικά λάθη. Να τα εντοπίσετε και να τα διορθώσετε, συμπληρώνοντας τον αριθμό της γραμμής στην οποία βρίσκεται κάθε λάθος και τη σωστή εντολή, στον πίνακα απαντήσεων που σας δίνεται. Δεν επιτρέπεται καμία προσθήκη ή αφαίρεση εντολής στο πρόγραμμα.

```
/*1*/ #include <iostream>
/*2*/ #define N 100
/*3*/ using namespace std;
/*4*/ int main( ) {
/*5*/ int Grades[ ], i;
/*6*/ string Names[N];
/*7*/ for (i = 1; i < N; i++) {
/*8*/     cout << "Δώσε όνομα και βαθμό μαθητή: "<<endl;
/*9*/     cin >> Names[i] >> Grades[i]; }
/*10*/ int minGrade = Grades[0];
/*11*/ for (i = 1; i < N; i++)
/*12*/     if (Grades[i] > minGrade)
/*13*/         minGrade = Grades[i];
/*14*/ cout << "Οι μαθητές με τον μικρότερο βαθμό="<< minGrade<< " είναι:"<<endl;
/*15*/ for (i = 0; i < N; i++)
/*16*/     if (Grades[i] = minGrade)
/*17*/         cout << Names[i] << endl;
/*18*/ return 0; }
```

A/A	Αριθμός Γραμμής	Διορθωμένη Εντολή
1	5	int Grades[N], i;
2	7	for (i = 0; i < N; i++) {
3	12	if (Grades[i] < minGrade)
4	16	if (Grades[i] == minGrade)

1,25 μονάδες ο σωστός εντοπισμός του λάθους και 1,25 μονάδες η σωστή διορθωμένη εντολή.

8. Δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο διαβάζει το όνομα και την ηλικία (ακέραιος αριθμός) 1000 κατοίκων μιας πόλης. Το πρόγραμμα ελέγχει ότι η ηλικία που εισάγεται είναι μεταξύ 1 και 120. Σε αντίθετη περίπτωση, εμφανίζει μήνυμα λάθους και ζητά εκ νέου από τον χρήστη να εισάγει όνομα και ηλικία. Στη συνέχεια, το πρόγραμμα υπολογίζει και εμφανίζει το πλήθος των ατόμων που έχουν δικαίωμα ψήφου στις εκλογές. Ένα άτομο έχει δικαίωμα ψήφου εάν η ηλικία του είναι μεγαλύτερη από 17 ετών. Τέλος, το πρόγραμμα εντοπίζει τον μεγαλύτερο σε ηλικία ψηφοφόρο και εμφανίζει το όνομά του στην οθόνη.

```
#include <iostream>
#define N 1000
using namespace std;
int main()
{
    
    for (i=0;i<N;i++)
    {
        cout << "Δώσε όνομα και ηλικία: ";
        cin >> name >> age;

        while (  )
        {
            cout << "Λάθος τιμή. Δώσε ξανά όνομα και ηλικία: ";
            cin >> name >> age;
        }

        

        if (age > 17 && age > maxAge)
        {
            
        }
    }
    cout << "Πλήθος ψηφοφόρων: " << voters;
    cout << "Μεγαλύτερος σε ηλικία ψηφοφόρος: " << maxName;
```

```
return 0;  
}
```

Να γράψετε:

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

int age,voters=0,maxAge=0, i; 1.25 μονάδες
string name,maxName; 1.25 μονάδα
Σύνολο 2.5 μονάδες

(β) Τη συνθήκη που πρέπει να τοποθετηθεί στη θέση Β η οποία θα ελέγχει πως οι τιμές της ηλικίας που δίνονται είναι ορθές (μεταξύ 1 και 120 συμπεριλαμβανομένων). Στην περίπτωση που οι τιμές είναι λανθασμένες τυπώνεται το αντίστοιχο μήνυμα και ο χρήστης ξαναδίνει τιμές.

age < 1 || age >120
2.5 μονάδες η ορθή εντολή

(γ) Τις εντολές που πρέπει να τοποθετηθούν στη θέση Γ ώστε να υπολογίζεται το πλήθος των ψηφοφόρων.

if (age > 17)
voters++;
1.25 μονάδες η κάθε σωστή εντολή, σύνολο 2.5 μονάδες

(δ) Τις εντολές που πρέπει να τοποθετηθούν στη θέση Δ ώστε να υπολογίζεται ο μεγαλύτερος σε ηλικία ψηφοφόρος.

maxAge = age;

maxName = name;

1.25 μονάδες η κάθε σωστή εντολή, σύνολο 2.5 μονάδες

9. Σας δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο διαβάζει από το αρχείο input.txt στοιχεία άγνωστου πλήθους προϊόντων μιας αποθήκης. Για κάθε προϊόν δίνονται ο κωδικός του και η ποσότητά του σε απόθεμα. Ακολουθως, εντοπίζει και τυπώνει στο αρχείο prod200.txt, τους κωδικούς και τις ποσότητες εκείνων των προϊόντων που έχουν ποσότητα μεγαλύτερη από 200. Επίσης εντοπίζει και τυπώνει στο αρχείο prod50.txt, τους κωδικούς και τις ποσότητες εκείνων των προϊόντων που έχουν ποσότητα μικρότερη από 50.

Παράδειγμα Εισόδου (input.txt) (Παράδειγμα για 5 προϊόντα) 1001 250 2008 13 3300 225 4107 42 5600 160	Παράδειγμα Εξόδου (prod200.txt) 1001 250 3300 225 Παράδειγμα Εξόδου (prod50.txt) 2008 13 4107 42
--------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------

Δίνεται το ακόλουθο πρόγραμμα:

```
#include <fstream>
using namespace std;
int main() {
  A
  ofstream fout50("prod50.txt");
  int code, quantity;
  while ( B ) {
    Γ
    if (quantity > 200) {
      Δ1
    }
    if (quantity < 50) {
      Δ2
    }
  }
  fin.close();
  fout200.close();
  fout50.close();
  return 0;
}
```

Να γράψετε:

(α) Τις εντολές που πρέπει να τοποθετηθούν στη θέση Α, έτσι ώστε να δημιουργηθούν οι κατάλληλες ροές ανάγνωσης και εγγραφής από και προς τα αρχεία.

ifstream fin("input.txt");

1.25 μονάδες

ofstream fout200("prod200.txt");

1.25 μονάδες

Σύνολο 2.5 μονάδες

(β) Τη συνθήκη που πρέπει να τοποθετηθεί στη θέση Β για άγνωστο πλήθος στοιχείων στο αρχείο.

!fin.eof()

2.5 μονάδες η ορθή απάντηση

(γ) Την εντολή που πρέπει να τοποθετηθεί στη θέση Γ, για ανάγνωση των κατάλληλων στοιχείων από το αρχείο.

fin >> code >> quantity

2.5 μονάδες η ορθή απάντηση

(δ) Τις εντολές που πρέπει να τοποθετηθούν στη θέση Δ1 και Δ2, για κατάλληλη εγγραφή στα αρχεία (όπως φαίνεται στα παραδείγματα εξόδου).

Δ1: fout200 << code << " " << quantity << endl; 1.25 μονάδες η ορθή απάντηση

Δ2: fout50 << code << " " << quantity << endl; 1.25 μονάδες η ορθή απάντηση

Σύνολο 2.5 μονάδες

10. Να γράψετε πρόγραμμα στη γλώσσα προγραμματισμού C++ το οποίο να εκτελεί τα παρακάτω:

(α) Να δημιουργήσετε συνάρτηση με όνομα comp η οποία δέχεται έναν ακέραιο αριθμό και στη συνέχεια υπολογίζει και επιστρέφει στο κύριο πρόγραμμα (main) το παραγοντικό του αριθμού όπως επίσης και το διπλό παραγοντικό του.

Το παραγοντικό ενός αριθμού N συμβολίζεται με θαυμαστικό N! και υπολογίζεται πολλαπλασιάζοντας όλους τους θετικούς ακέραιους της ακολουθίας από το 1 μέχρι το N:

$$N! = 1 * 2 * 3 \dots N$$

Παράδειγμα: $5! = 1 * 2 * 3 * 4 * 5 = 120$

Το διπλό παραγοντικό ενός αριθμού N συμβολίζεται με N!! και υπολογίζεται πολλαπλασιάζοντας όλους τους θετικούς ακέραιους της ακολουθίας από το N μέχρι το 1 ανά δύο:

$$N!! = N * (N - 2) * (N - 4) \dots$$

Παράδειγμα: $5!! = 5 * 3 * 1 = 15$

(β) Στο κύριο πρόγραμμα (main) να διαβάζονται δύο ακέραιοι αριθμοί x, y. Με τη χρήση της συνάρτησης comp να υπολογίζεται η τιμή της εξίσωσης $z = x! + y! + x!! + y!!$

Η τιμή της εξίσωσης να εμφανίζεται από το κύριο πρόγραμμα (main).

```
#include <iostream>
using namespace std;
```

```
void comp(int N, int &fact, int &dfact) {
    fact = 1;
    for (int i = 1; i <= N; i++) {
        fact *= i;
    }

    dfact = 1;
    for (int i = N; i >= 1; i -= 2) {
        dfact *= i;
    }
}
```

```
int main()
{
    int x, y;
    int fx, fy;
    int dfx, dfy;

    cin >> x >> y;
```

```
    comp(x, fx, dfx);
    comp(y, fy, dfy);
```

```
    int z = fx + fy + dfx + dfy;
    cout << z << endl;
    return 0;
```

```
}
```

Η λύση είναι ενδεικτική. Οποιαδήποτε άλλη επιστημονικά τεκμηριωμένη λύση είναι αποδεκτή.

Αφαιρετικά γενικά

- 0.25 αν λείπει το `#include<iostream>`
- 0.25 αν λείπει το `using namespace std;`
- 0.25 αν λείπει το `int main()`
- 0.25 αν λείπει { ή και }
- 0.25 αν λείπουν δύο ή περισσότερα ; (μια φορά)
- 0.25 αν λείπει το `return 0;`

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

11. Στον 1ο Παγκύπριο Διαγωνισμό Τεχνικών Δεξιοτήτων Ρομποτικής του κλάδου Ηλεκτρολογίας και Ηλεκτρονικών Εφαρμογών στη διαγωνιστική κατηγορία ΝΑΟ, λαμβάνουν μέρος 15 ομάδες από όλες τις τεχνικές σχολές της Κύπρου. Κάθε ομάδα καλείται να ρυθμίσει και να προγραμματίσει το ρομπότ της για την αντιμετώπιση και ολοκλήρωση μίας πρόκλησης. Η πρόκληση βαθμολογείται από 1 έως 100 ακέραιες μονάδες. Επιπρόσθετα καταγράφεται ο χρόνος ολοκλήρωσης της πρόκλησης. Ο χρόνος μιας ομάδας μηδενίζεται αν η ομάδα δεν ολοκληρώσει την πρόκληση. Τέλος, αν μια ομάδα παραβεί κάποιον από τους όρους του διαγωνισμού, τότε η βαθμολογία της μηδενίζεται.

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

(α) Να χρησιμοποιεί τη συνάρτηση `ReadData`, η οποία θα καταχωρεί:

- i. Στον πίνακα `Teams` τα ονόματα των ομάδων.
- ii. Στον πίνακα `Scores` τη βαθμολογία της πρόκλησης για κάθε ομάδα.
- iii. Στον πίνακα `Violation` την ένδειξη παράβασης (1 στην περίπτωση μη παράβασης ή 0 στην περίπτωση παράβασης) για κάθε ομάδα.
- iv. Στον πίνακα `Time` τον χρόνο ολοκλήρωσης της πρόκλησης για κάθε ομάδα.

(β) Να χρησιμοποιεί τη συνάρτηση `TotalScore`, η οποία θα δέχεται ως παραμέτρους τους πιο πάνω πίνακες και θα υπολογίζει την τελική βαθμολογία κάθε ομάδας ως εξής: αν υπάρχει παράβαση ή δεν ολοκληρωθεί η πρόκληση η βαθμολογία μηδενίζεται, αλλιώς διαιρείται με τον χρόνο ολοκλήρωσης της πρόκλησης της κάθε ομάδας. Το αποτέλεσμα για κάθε ομάδα αποθηκεύεται στον πίνακα `Totals`.

(γ) Να χρησιμοποιεί τη συνάρτηση `Results` για να εμφανίζει αλλά και να αποθηκεύει σε αρχείο κειμένου (π.χ. "results.txt") τα ονόματα των ομάδων και τις τελικές τους βαθμολογίες.

Σημείωση: Το πρόγραμμα πρέπει να εμφανίζει κατάλληλα μηνύματα για την εισαγωγή των δεδομένων και την παρουσίαση των αποτελεσμάτων.

```

#include <iostream>
#include <fstream>
using namespace std;
#define N 15
void ReadData(string Teams[ ], int Scores[ ], int Violation[ ], float Time[ ])
{
    for (int i = 0; i < N; i++) {
        cout << endl << "ΟΜΑΔΑ " << i+1 << endl;
        cout << "Όνομα ομάδας: ";
        cin >> Teams[i];
        cout << "Βαθμολογία: ";
        cin >> Scores[i];
        cout << "Παράβαση (1=ΟΧΙ, 0=ΝΑΙ): ";
        cin >> Violation[i];
        cout << "Χρόνος: ";
        cin >> Time[i];
        cout << endl;
    }
}
void TotalScore(int Scores[ ], int Violation[ ], float Time[ ], float Totals[ ]) {
    for (int i = 0; i < N; i++) {
        if (Violation[i] == 0 || Time[i] == 0)
        {
            Totals[i]=0;
        }
        else
        {
            Totals[i] = (float)Scores[i] / Time[i];
        }
    }
}

```

3 μονάδες

3 μονάδες

```

void Results(string Teams[ ], float Totals[ ]) {
    ofstream file("results.txt");
    cout << endl << "--- ΑΠΟΤΕΛΕΣΜΑΤΑ ---" << endl;
    for (int i = 0; i < N; i++) {
        cout << Teams[i] << " - " << Totals[i] << endl;
        file << Teams[i] << " - " << Totals[i] << endl;
    }
    file.close();
}

int main() {
    string Teams[N];
    int Scores[N], Violation[N];
    float Time[N], Totals[N];
    ReadData(Teams, Scores, Violation, Time);
    TotalScore(Scores, Violation, Time, Totals);
    Results(Teams, Totals);
    return 0;
}

```

3 μονάδες

1.5 μονάδες

1.5 μονάδες

Κάθε σωστή και ολοκληρωμένη συνάρτηση στα τρία ερωτήματα: 3 μονάδες

Ο σωστός ορισμός των μεταβλητών στην κύρια συνάρτηση: 1.5 μονάδες

Οι σωστές εντολές για την κλήση των συναρτήσεων στην κύρια συνάρτηση: 1.5 μονάδες

Αφαιρετικά (γενικά)

- 0.25 αν λείπει το #include <iostream>
- 0.25 αν λείπει το #include <fstream>
- 0.25 αν λείπει το using namespace std;
- 0.25 αν λείπει το int main ()
- 0.25 αν λείπει { ή και }
- 0.25 αν λείπουν δύο ή περισσότερα ; (μια φορά)
- 0.25 αν λείπει το return 0;

Σημείωση: Η λύση είναι ενδεικτική. Οποιαδήποτε άλλη επιστημονικά τεκμηριωμένη λύση είναι αποδεκτή.