

ΕΡΓΑΣΙΑ ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ II

ΤΜΗΜΑ: ΤΕΧΝΙΚΟΣ ΕΦΑΡΜΟΓΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Β΄
ΜΑΘΗΜΑ: ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ II
ΚΑΤΑΡΤΙΖΟΜΕΝΟΣ/Η: Αριστοτελης Γκινης
ΕΚΠΑΙΔΕΥΤΕΣ: ΠΛΥΤΑ Ε. - ΤΣΙΚΕΡΔΗΣ Δ.
ΗΜ/ΝΙΑ ΠΑΡΑΔΟΣΗΣ ΕΡΓΑΣΙΑΣ: 05/04/2022

Περιεχόμενα

ΠΕΡΙΕΧΟΜΕΝΑ.....	1
ΘΕΜΑ 1 ^ο	2
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	3
ΘΕΜΑ 2 ^ο	4
Άσκηση 1 ^η	4
Άσκηση 2 ^η	6
Κώδικας :.....	6
Άσκηση 3 ^η	7
Κώδικας :.....	7
Άσκηση 4 ^η	8
Κώδικας :.....	8
Άσκηση 5 ^η	10
Κώδικας :.....	10
Άσκηση 6 ^η	11
Κώδικας :.....	11
Άσκηση 7 ^η	13
Κώδικας :.....	13
Άσκηση 8 ^η	13
Κώδικας :.....	13
Άσκηση 9 ^η	13
Κώδικας :.....	13
Άσκηση 10 ^η	14
Κώδικας :.....	14

ΤΙΤΛΟΣ

**Αντικειμενοστραφής Προγραμματισμός – Διαφορές από τον διαδικαστικό προγραμματισμό
– Πλεονεκτήματα/ Μειονεκτήματα. Για ποιους λόγους κυριάρχησε η Java στις
αντικειμενοστραφείς γλώσσες προγραμματισμού.**

Στην πληροφορική Αντικειμενοστρεφή προγραμματισμό (ΑΠ) ή Object-Oriented Programming (OOP), ονομάζουμε ένα προγραμματιστικό υπόδειγμα το οποίο εμφανίστηκε στα τέλη της δεκαετίας του 1960 και καθιερώθηκε κατά τη δεκαετία του 1990, αντικαθιστώντας σε μεγάλο βαθμό το παραδοσιακό υπόδειγμα του διαδικαστικού προγραμματισμού.

Στην πληροφορική διαδικαστικός προγραμματισμός (procedural programming) είναι μία προσέγγιση στον προγραμματισμό, η οποία βασίζεται στην έννοια της *κλήσης διαδικασίας*. Η διαδικασία, γνωστή επίσης και ως ρουτίνα, υπορουτίνα, μέθοδος ή συνάρτηση, είναι απλά ένα αυτοτελές σύνολο εντολών προς εκτέλεση.

Η βασική διαφορά μεταξύ του Αντικειμενοστρεφή και του Διαδικαστικού Προγραμματισμού είναι ότι το επίκεντρο του Διαδικαστικού Προγραμματισμού είναι να αναλύσει την εργασία προγραμματισμού σε μια συλλογή μεταβλητών και υπορουτίνων ενώ, το επίκεντρο του Αντικειμενοστρεφή προγραμματισμού είναι να αναλύσει την εργασία προγραμματισμού σε αντικείμενα, τα οποία ενσωματώνουν δεδομένα και μεθόδους, έτσι ο Αντικειμενοστρεφής προγραμματισμός είναι πιο εύχρηστος ενώ ταυτόχρονα μπορεί να προσομοιώσει προβλήματα του πραγματικού κόσμου με την χρήση αντικειμένων. Βεβαία τα αρχεία των προγραμμάτων που έχουν σχεδιαστεί με Αντικειμενοστρεφή προγραμματισμός πιάνουν περισσότερο χώρο στην μονάδα αποθήκευσης. Είναι σημαντικό να αναφερθεί πως ο ΑΠ απαιτεί περισσότερες γνώσεις από των χρήστη αφού πρώτα απ όλα πρέπει να κατανοήσει την έννοια των αντικειμένων .

Κεντρική ιδέα στον αντικειμενοστρεφή προγραμματισμό είναι η κλάση (class), μία αυτοτελής και αφαιρετική αναπαράσταση κάποιας κατηγορίας αντικειμένων, είτε φυσικών είτε νοητών, εννοιολογικών αντικειμένων, σε ένα περιβάλλον προγραμματισμού. Πρακτικά είναι ένας τύπος δεδομένων, ή αλλιώς το προσχέδιο μίας δομής δεδομένων με δικά της περιεχόμενα, τόσο μεταβλητές όσο και διαδικασίες. _Οι διαδικασίες των κλάσεων συνήθως καλούνται μέθοδοι (methods) και οι μεταβλητές τους γνωρίσματα (attributes) ή πεδία (fields). Το αντικείμενο (object) είναι το στιγμιότυπο μίας κλάσης, δηλαδή αυτή καθαυτή η δομή δεδομένων (με αποκλειστικά δεσμευμένο χώρο στη μνήμη) βασισμένη στο «καλούπι» που προσφέρει η κλάση. Στον Αντικειμενοστρεφή προγραμματισμό υπάρχουν τέσσερις βασικές αρχές Ενθυλάκωση (Encapsulation), Πολυμορφισμός (Polymorphism), Κληρονομικότητα (Inheritance) και Αφηρημένη κλάση (Abstraction).

Η Ενθυλάκωση δεδομένων είναι η ιδιότητα που προσφέρουν οι κλάσεις να «κρύβουν» τα ιδιωτικά δεδομένα τους από το υπόλοιπο πρόγραμμα και να εξασφαλίζουν πως μόνο μέσω των δημόσιων μεθόδων τους θα μπορούν αυτά να προσπελαστούν.

Ο Πολυμορφισμός επιτρέπει το χειρισμό τιμών διαφορετικών τύπων δεδομένων με χρήση μιας ομοιόμορφης διεπαφής. Μια συνάρτηση που μπορεί να αποτιμηθεί ή να εφαρμοστεί σε τιμές διαφορετικών τύπων είναι γνωστή ως πολυμορφική συνάρτηση και ένας τύπος δεδομένων που εμφανίζεται ως γενικευμένου τύπου ονομάζεται πολυμορφικός τύπος δεδομένων

Η Κληρονομικότητα ονομάζεται η ιδιότητα των κλάσεων να επεκτείνονται σε νέες κλάσεις, ρητά δηλωμένες ως κληρονόμους (*υποκλάσεις* ή 'θυγατρικές κλάσεις'), οι οποίες μπορούν να επαναχρησιμοποιήσουν τις μεταβιβάσιμες μεθόδους και ιδιότητες της γονικής τους κλάσης

αλλά και να προσθέσουν δικές τους. Αντικείμενα των θυγατρικών κλάσεων μπορούν να χρησιμοποιηθούν όπου απαιτούνται αντικείμενα των γονικών όμως το αντίστροφο δεν ισχύει. Η Αφηρημένη κλάση είναι μία κλάση που ορίζεται μόνο για να κληρονομηθεί σε θυγατρικές υποκλάσεις και δεν υπάρχουν δικά της αντικείμενα.

Η Java είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού που σχεδιάστηκε το 1995 από την εταιρεία πληροφορικής *Sun Microsystems* η οποία εξαγοράστηκε από την εταιρία λογισμικού *Oracle* το 2009. Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε *Java* τρέχουν σε Windows, Linux, Android, Macintosh κ.α χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση. Η java προσφέρει μια εμπειρία φιλική προς τον χρήστη, με απλο συντακτικό που πλυσιαζει την αγγλικη γλωσσα και με πολλά άλλα εργαλεία που βοηθούν στην χρήση της , επίσης υποστηρίζεται από έναν μεγάλο αριθμό βιβλιοθηκών, ενώ ταυτόχρονα υπάρχουν οδηγίες (documentation) που εξηγούν τις λειτουργίες της γλώσσας. Πολύ σημαντικό είναι πως υπερτερεί σε σχέση με άλλες γλώσσες όπως η C++ στην δημιουργία εφαρμογών λόγω της δομής της αλλά και το ότι προσφέρει μεγαλύτερη ασφάλεια. Τέλος αν και η Java είναι ενεργή πάνω από 25 χρονιά φαίνεται πως έχει ακόμα αρκετό καιρό μπροστά της.

ΒΙΒΛΙΟΓΡΑΦΙΑ

https://el.wikipedia.org/wiki/Αντικειμενοστρεφής_προγραμματισμός
<https://eclass.hmu.gr/modules/document/index.php?course=TP183&download=/4b975e3f96ya/4f61e6bev66/4f61e803776l.pdf>
<https://el.strephonsays.com/object-oriented-programming-and-vs-procedural-programming-13453>
<https://www.theserverside.com/feature/5-reasons-why-Java-is-still-the-best-programming-language>
<https://el.wikipedia.org/wiki/Java>
<https://www.businessinsider.com/oracle-bought-sun-because-of-ibm-not-google-2016-5>
<https://www.geeksforgeeks.org/benefits-advantages-of-oop/>

Άσκηση 1^η**Κώδικας :**

```

#include <iostream>
//Aristotelis Gkinis exercise.1
using namespace std;

int main()
{
    string currency;
    double ammountOM;
    double exchange = 0;
    double gbpSum = 0;
    double usdSum = 0;
    double rubSum = 0;
    double ypySum = 0;
    double overalSum = 0;

    while(ammountOM != 0 && currency != "x") {

        cout<< "**Enter x to exit* \n";
        cout<< "Enter type of currency \n";
        cout<< "(GBP, USD, RUB or JPY):\n";
        cin>> currency;

        if (currency == "GBP") {
            cout<< "Now Enter the ammount you want to convert: ";
            cin>> ammountOM;
            exchange = ammountOM * 1.18;
            gbpSum += exchange;

            cout<<"-----\n";
            cout <<ammountOM << " GBP are: " << exchange << " Euros."<< endl;
            cout<<"-----\n";
        }
        else if (currency == "USD"){
            cout<< "Now Enter the ammount you want to convert: ";
            cin>> ammountOM;
            exchange = ammountOM * 0.9;
            usdSum += exchange;

            cout<<"-----\n";
            cout <<ammountOM << " USD are: " << exchange << " Euros."<< endl;
            cout<<"-----\n";
        }
        else if (currency == "RUB"){
            cout<< "Now Enter the ammount you want to convert: ";
            cin>> ammountOM;
            exchange = ammountOM * 0.01;
            rubSum += exchange;

            cout<<"-----\n";
            cout <<ammountOM << " RUB are: " << exchange << " Euros."<< endl;
            cout<<"-----\n";
        }
        else if (currency == "JPY"){
            cout<< "Now Enter the ammount you want to convert: ";
            cin>> ammountOM;
            exchange = ammountOM * 0.0074;
            ypySum += exchange;

            cout<<"-----\n";

```

```

        cout << ammountOM << " JPY are: " << exchange << " Euros." << endl;
        cout << "-----\n";

    } else {
        cout << "-----\n";
        cout << "Invalid currency. Try Again:\n";
        cout << "-----\n";
    }

    overallSum += exchange;
    exchange = 0;

    cout << "The sum of the exchanges from GBP to EU is: " << gbpSum << endl;
    cout << "-----\n";
    cout << "The sum of the exchanges from USD to EU is: " << usdSum << endl;
    cout << "-----\n";
    cout << "The sum of the exchanges from RUB to EU is: " << rubSum << endl;
    cout << "-----\n";
    cout << "The sum of the exchanges from JPY to EU is: " << ypySum << endl;
    cout << "-----\n";
    cout << "The overall sum is: " << overallSum << endl;
    cout << "-----\n";

}

if (currency == "x" || ammountOM == 0) {

    cout << "Terminating...\n";
    cout << "-----\n";

}

return 0;
}

```

Αποτέλεσμα :

```

*Enter x to exit*
Enter type of currency
(GBP, USD, RUB or JPY):
USD
Now Enter the ammount you want to convert: 60
-----
60 USD are: 54 Euros.
-----
The sum of the exchanges from GBP to EU is: 0
-----
The sum of the exchanges from USD to EU is: 54
-----
The sum of the exchanges from RUB to EU is: 0
-----
The sum of the exchanges from JPY to EU is: 0
-----
The overall sum is: 54
-----
*Enter x to exit*
Enter type of currency
(GBP, USD, RUB or JPY):
USD
Now Enter the ammount you want to convert: 8
-----
8 USD are: 7.2 Euros.
-----
The sum of the exchanges from GBP to EU is: 0
-----
The sum of the exchanges from USD to EU is: 61.2
-----
The sum of the exchanges from RUB to EU is: 0
-----
The sum of the exchanges from JPY to EU is: 0
-----
The overall sum is: 61.2
-----
*Enter x to exit*
Enter type of currency
(GBP, USD, RUB or JPY):
GBP
Now Enter the ammount you want to convert: 6
-----
6 GBP are: 7.08 Euros.
-----
The sum of the exchanges from GBP to EU is: 7.08
-----
The sum of the exchanges from USD to EU is: 61.2
-----
The sum of the exchanges from RUB to EU is: 0
-----
The sum of the exchanges from JPY to EU is: 0
-----
The overall sum is: 68.28

```

Άσκηση 2^η

Κώδικας :

```
#include <iostream>
//Aristotelis Gkinis exercise.2
using namespace std;

int main()
{
    char destination;
    char mOT;
    int seats;
    double money = 0;
    double sum = 0;
    double italySum = 0;
    double airTicketsSum = 0;

    while(seats != 0) {
        money = 0;
        cout<< "Enter your destination: \n";
        cout<< "-Italy(I), Cyprus(C) or Turkey(T): ";
        cin>> destination;

        cout<< "Enter means of transport:\n";
        cout<< "-Airplane(A) or Ship(S): ";
        cin>> mOT;

        cout<< "Enter 0 to exit \n";
        cout<< "Enter number of seats: ";
        cin>> seats;

        switch(mOT){

            case 'A':
                if(destination == 'I') {
                    money = seats * 300;
                    italySum += money;
                } else if (destination == 'C') {
                    money = seats * 350;
                } else if (destination == 'T') {
                    money = seats * 250;
                } else {
                    cout<<"-----\n";
                    cout<< "***Not available destination\n Please choose I, C or T**\n";
                }
                airTicketsSum += money;
                sum += money;
                break;

            case 'S':
                if(destination == 'I') {
                    money = seats * 200;
                    italySum += money;
                } else if (destination == 'C') {
                    money = seats * 250;
                } else if (destination == 'T') {
                    money = seats * 150;
                } else {
                    cout<<"-----\n";
                    cout<< "***Not available destination\n Please choose I, C or T**\n";
                }
                sum += money;
                break;
        }
    }
}
```

```

        default :
            cout<< " Not available means of transportation\n";
        }

        cout<<"-----\n";
        cout<<"The price is "<<money<<" EURO"<<endl;
        cout<<"-----\n";
        cout<<"The sum is "<<sum<<" EURO"<<endl;
        cout<<"The sum of the tickets for Itay is "<<italySum<<" EURO"<<endl;
        cout<<"The air tickets sum is "<<airTicketsSum<<" EURO"<<endl;

        cout<<"-----\n";
    }

    return 0;
}

```

Αποτέλεσμα :

```

Enter your destination:
-Itay(I), Cyprus(C) or Turkey(T)): I
Enter means of transport:
-Airplane(A) or Ship(S): A
Enter 0 to exit
Enter number of seats: 2
-----
The price is 600 EURO
-----
The sum is 600 EURO
The sum of the tickets for Itay is 600 EURO
The air tickets sum is 600 EURO
-----
Enter your destination:
-Itay(I), Cyprus(C) or Turkey(T)): I
Enter means of transport:
-Airplane(A) or Ship(S): S
Enter 0 to exit
Enter number of seats: 1
-----
The price is 200 EURO
-----
The sum is 800 EURO
The sum of the tickets for Itay is 800 EURO
The air tickets sum is 600 EURO
-----
Enter your destination:
-Itay(I), Cyprus(C) or Turkey(T)): T
Enter means of transport:
-Airplane(A) or Ship(S): S
Enter 0 to exit
Enter number of seats: 0
-----
The price is 0 EURO
-----
The sum is 800 EURO
The sum of the tickets for Itay is 800 EURO
The air tickets sum is 600 EURO
-----

```

Άσκηση 3^η

Κώδικας :

```

#include <iostream>
//Aristotelis Gkinis exercise.3
using namespace std;

int main()
{
    int volume;
    int taxValue;
    int tax[9];

    for (int i = 0; i <= 9; i++) {
        cout<<"Enter the volume of the Car NO."<< i+1 << endl;
        cin>> volume;
        if(volume >= 0 && volume <= 1100) {
            tax[i] = 110;
            cout<<endl;

        } else if(volume >= 1101 && volume <= 1400) {
            tax[i] = 150;
            cout<<endl;
        }
    }
}

```

```

    } else if(volume >= 1401 && volume <= 2000) {
        tax[i] = 225;
        cout<<endl;

    } else if(volume >= 2001) {
        tax[i] = 600;
        cout<<endl;
    }

}

cout<<"-----\n";
for (int i = 0; i <= 9; i++) {
    cout<<"The taxes for the Car NO."<<i+1 << " is : "<<tax[i]<<" Euro"<<endl;
}

return 0;
}

```

Αποτέλεσμα :

```

Enter the volume of the Car NO.1
1000
Enter the volume of the Car NO.2
3000
Enter the volume of the Car NO.3
2000
Enter the volume of the Car NO.4
1200
Enter the volume of the Car NO.5
1500
Enter the volume of the Car NO.6
1660
Enter the volume of the Car NO.7
1800
Enter the volume of the Car NO.8
5000
Enter the volume of the Car NO.9
1250
Enter the volume of the Car NO.10
2080

-----
The taxes for the Car NO.1 is : 110 Euro
The taxes for the Car NO.2 is : 600 Euro
The taxes for the Car NO.3 is : 225 Euro
The taxes for the Car NO.4 is : 150 Euro
The taxes for the Car NO.5 is : 225 Euro
The taxes for the Car NO.6 is : 225 Euro
The taxes for the Car NO.7 is : 225 Euro
The taxes for the Car NO.8 is : 600 Euro
The taxes for the Car NO.9 is : 150 Euro
The taxes for the Car NO.10 is : 600 Euro

```

Άσκηση 4"

Κώδικας :

```

#include <iostream>
// Aristotelis Gkinis exercise.4
#define UNDERLINE "\033[4m"

#define CLOSEUNDERLINE "\033[0m"

using namespace std;

int main()
{
    int airPollution[5];
    int pollutionValue = 0;
    int countDanger = 0;
    int sum = 0;
    double averagePolution = 0;

    for (int i = 0; i <= 5; i++) {
        cout<<"Enter the pollution status"<<" of Day "<<i+1<<": ";
        cin>> pollutionValue;
        airPollution[i] = pollutionValue;
    }
}

```



```

cout<<"-----\n";

for (int i = 0; i<= 5; i++) {

    if(airPollution[i] <= 1 && airPollution[i] >= 0) {

        cout<<"Day "<<i+1<<" pollution: "<<airPollution[i]<< endl;
        cout<<"Normal air pollution\n";
    } else if(airPollution[i] == 2) {
        cout<<"Day "<<i+1<<" pollution: "<<airPollution[i]<< endl;
        cout<<"Air pollution at the limit\n";

    } else if(airPollution[i] > 2) {

        cout<<"Day "<<i+1<<" pollution: "<<airPollution[i]<< endl;
        cout<<"Dangerous air pollution\n";

        if(airPollution[i] > 3) {
            countDanger++;
        }

    } else {
        cout<<"Invalid pollution value\n";
    }

    cout<<"-----\n";

}

for (int i = 0; i<= 5; i++) {
    sum += airPollution[i];
    averagePolution = sum / 30;

}

cout<< endl;
cout<<UNDERLINE<< "Months informations" <<CLOSEUNDERLINE<<endl<<endl;
cout<<"-The average pollution of the month is : "<<averagePolution<<endl;
cout<<"-Number of dangerous days(polution over 3): "<<countDanger<<endl;

return 0;
}

```

Αποτέλεσμα :

```

Enter the pollution status of Day 1: 3
Enter the pollution status of Day 2: 2
Enter the pollution status of Day 3: 4
Enter the pollution status of Day 4: 1
Enter the pollution status of Day 5: 0
Enter the pollution status of Day 6: 1
-----
Day 1 pollution: 3
Dangerous air pollution
-----
Day 2 pollution: 2
Air pollution at the limit
-----
Day 3 pollution: 4
Dangerous air pollution
-----
Day 4 pollution: 1
Normal air pollution
-----
Day 5 pollution: 0
Normal air pollution
-----
Day 6 pollution: 1
Normal air pollution
-----

Months informations

-The average pollution of the month is :0
-Number of dangerous days(pollution over 3): 1

```

Άσκηση 5"

Κώδικας :

```

#include <iostream>
// Aristotelis Gkinis exercise.5
#define UNDERLINE "\033[4m"

#define CLOSEUNDERLINE "\033[0m"

using namespace std;

int main()
{
    int viewersAge[4];
    int age = 0;
    int sum = 0;
    int countChild = 0;
    int countElder = 0;

    cout<<UNDERLINE<<"*****Tickets shop*****"<<CLOSEUNDERLINE<<endl<<endl;

    cout<<"-----\n";
    cout<<"Tickets price for children until the age of 12 is 8 Euros\n"
        "Tickets price for viewers from the age of 12 to 65 is 15 Euros\n"
        "Tickets price for viewers over 65 years old is 10 Euros\n";
    cout<<"-----\n\n";

    for (int i = 0; i<= 4; i++) {

        cout<<"Enter the age of the viewer "<<i+1<<": ";
        cin>> age;
        viewersAge[i] = age;

    }

    cout<<"\n-----\n";

    for (int i = 0; i<= 4; i++) {

        int ticketPrice = 0;
        if(viewersAge[i] <= 12 ) {
            countChild++;

```

```

        ticketPrice = 8;
        cout<<"Viewer NO."<<i+1<<" ticket price: "<<ticketPrice<<"Euros"<< endl;
    } else if(viewersAge[i] >= 12 && viewersAge[i] <= 65) {
        ticketPrice = 15;
        cout<<"Viewer NO."<<i+1<<" ticket price: "<<ticketPrice<<"Euros"<< endl;
    } else if(viewersAge[i] > 65) {
        countElder++;
        ticketPrice = 10;
        cout<<"Viewer NO."<<i+1<<" ticket price: "<<ticketPrice<<"Euros"<< endl;
    } else {
        cout<<"Invalid age\n";
    }

    sum += ticketPrice;

    cout<<"-----\n";
}

cout<<"\n===== \n";
cout<<"Total ticets receipts: "<< sum<<" Euros"<<endl;
cout<<"===== \n\n";

cout<<UNDERLINE<<"Viewers age informations" <<CLOSEUNDERLINE<<endl<<endl;
cout<<"-Children viewers: "<< countChild <<endl;
cout<<"-Elder viewers: "<< countElder <<endl;
cout<<"\n===== \n";

return 0;
}

```

Αποτέλεσμα :

```

*****Tickets shop*****

-----
Tickets price for children until the age of 12 is 8 Euros
Tickets price for viewers from the age of 12 to 65 is 15 Euros
Tickets price for viewers over 65 years old is 10 Euros
-----

Enter the age of the viewer 1: 9
Enter the age of the viewer 2: 25
Enter the age of the viewer 3: 68
Enter the age of the viewer 4: 70
Enter the age of the viewer 5: 10

-----
Viewer NO.1 ticket price: 8 Euros
-----
Viewer NO.2 ticket price: 15 Euros
-----
Viewer NO.3 ticket price: 10 Euros
-----
Viewer NO.4 ticket price: 10 Euros
-----
Viewer NO.5 ticket price: 8 Euros
-----

=====
Total ticets receipts: 51 Euros
=====

Viewers age informations
-Children viewers: 2
-Elder viewers: 2
=====

```

Άσκηση 6^η

Κώδικας :

```

#include <iostream>
#include <bits/stdc++.h>

```

```

// Aristotelis Gkinis exercise.6
#define UNDERLINE "\033[4m"

#define CLOSEUNDERLINE "\033[0m"
using namespace std;

int main() {

    string name = "";
    double grade;
    string maxName;
    int maxGrade = INT_MIN;
    double sum=0;
    double average = 0;
    int counts = 1;

    cout<<"*Enter x to get the results*"<<endl;
    cout<<"-----\n";

    while(true){

        cout<< "Enter name: ";
        cin>> name;
        if(name == "x") {

            cout<<"-----\n";
            break;

        }
        do{
            cout<< "Enter " << name << " grade: ";
            cin>> grade;
        } while (grade>20||grade<0);

        sum += grade;
        average = sum /counts;

        if(grade >= maxGrade) {
            maxGrade = grade;
            maxName = name;
        }

        cout<<"-----\n";
        counts++;

    }

    cout<< "Average: " << average <<
    "\n-----\n";

    cout<<UNDERLINE<< "Highest grade"<<CLOSEUNDERLINE<<endl<<endl;
    cout<<"-Name: " <<maxName <<
    "\n-Grade: " << maxGrade<<"\n-----\n";

    return 0;
}

```

Αποτέλεσμα :

```

*Enter x to get the results*
-----
Enter name: Tim
Enter Tim grade: 16
-----
Enter name: Jack
Enter Jack grade: 13
-----
Enter name: Bob
Enter Bob grade: 11
-----
Enter name: Mike
Enter Mike grade: 18
-----
Enter name: x
-----
Average: 14.5
-----
Highest grade
-----
-Name: Mike
-Grade: 18
-----

```

Άσκηση 7^η

Κώδικας :

Αποτέλεσμα :

Άσκηση 8^η

Κώδικας :

Αποτέλεσμα :

Άσκηση 9^η

Κώδικας :

```

#include <iostream>
#include <bits/stdc++.h>
//Aristotelis Gkinis exercise.9
using namespace std;

int findOddMin(int array[],int size);
int findEvenSum(int array[],int size);

int main() {
    int Pin[99];
    int number = INT_MIN;
    int count = 0;

    cout<<"-----\n";
    cout<<"Press 0 to EXIT\n";
    cout<<"Enter your password:\n";
    cout<<"-----\n";

    while(number != 0){
        cin>>number;
        Pin[count] = number;
        count++;
    }

    cout<<"-----\n";
    cout<<"The password has " <<count - 1<<" digits\n";
    cout<<"count :" << count<<endl;
    cout<<"Minimum number is :"<< findOddMin(Pin, count)<<endl;
    cout<<"The sum is :"<< findEvenSum(Pin, count)<<endl;

```

```

        cout<<"-----\n";

        return 0;
    }

    int findOddMin(int array[],int size) {

        int min = INT_MAX;

        for(int i=0; i<size-1; i++) {
            if( array[i] < min) {
                min = array[i];
            }
        }
        return min;
    }

    int findEvenSum(int array[],int size) {

        int sum = 0;

        for(int i=0; i<size-1; i++) {
            if( array[i] % 2 == 0) {
                sum += array[i];
            }
        }
        return sum;
    }
}

```

Αποτέλεσμα :

```

-----
Press 0 to EXIT
Enter your password:
-----
6
4
23
11
43
27
0
-----
The password has 6 digits
count :7
Minimum number is :4
The sum is :10
-----

```

Άσκηση 10^η

Κώδικας :

```

#include <iostream>
//Aristotelis Gkinis exercise.10
using namespace std;

void display(int array[],int size);
double findMo(int array[],int size);

int main()
{
    int number;
    int PRO[9];
    int mult = 0;
    cout << "Enter a number: ";
    cin>> number;
    cout<<"-----\n";
}

```

```

for(int i = 1; i <= 10 ; i++) {

    mult = number*i;
    PRO[i-1] = mult;
    cout << number << " * " << i << " = "<<mult << endl;

}

cout<<"-----\n";
cout<<"reversed: ";
display(PRO,9);
cout<<"\n-----\n";
cout<<"average: "<<findMo(PRO,9)<<endl;
cout<<"-----\n";

return 0;
}

void display(int array[],int size) {

    for(int i=size; i>=0; i--) {
        cout<<array[i]<<" ";
    }
}

double findMo(int array[],int size){
    int sum = 0;
    double average = 0;
    for(int i=0; i<=size; i++) {
        sum += array[i];
    }
    average = sum / (size+1);
    return average;
}

```

Αποτέλεσμα :

```

Enter a number: 6
-----
6 * 1 = 6
6 * 2 = 12
6 * 3 = 18
6 * 4 = 24
6 * 5 = 30
6 * 6 = 36
6 * 7 = 42
6 * 8 = 48
6 * 9 = 54
6 * 10 = 60
-----
reversed: 60 54 48 42 36 30 24 18 12 6
-----
average: 33
-----

```