Πρώτη προαιρετική

Γραψτε ένα πρόγραμμα σε C, όπου ο χρήστης να εισάγει τους πραγματικούς αριθμούς *a, b, c*. Αν υπάρχουν, το πρόγραμμα θα υπολογίζει και τυπώνει τις πραγματικές λύσεις του τριώνυμου:

$$ax^2 + bx + c = 0,$$

αλλίως θα αποκρίνεται ανάλογα.

Το πρόγραμμα οφείλει να μην καταρέει στις εξής 2 περίπτωσεις λανθασμένης είσοδου: (1) αν ο χρήστης εισάγει μη αναμενόμενες τιμές π.χ. (α = β = γ = 0), (2) αν η είσοδος περιέχει συντακτικά λάθη.

```
#include <stdio.h>

int
main()
{
        /* entolh ektypwshs */
        printf("Hello world\n");
        return 0;
}
```

```
#include <stdio.h>
#include <math.h>

int
main()
{
        double a, b, c, D, riza1, riza2;

        a = 5;
        b = 3;
        c = 9;

        D = b*b – 4*a*c;

        if (D < 0)
        {
                printf("No roots!\n");
        }
        else
        {
                riza1 = (-b + sqrt(D))/(2*a);
                riza2 = (-b - sqrt(D))/(2*a);
                printf("The roots of %f*x*x + %f*x + f = 0 are  %f and
                        %f\n",a,b,c,riza1,riza2);
        }

        return 0;
}
```

```
#include <stdio.h>

int
main()
{
      int a, b, c, result;

      a = 1;
      b = 1;

      c = a + b;

      printf("%d + %d = %d\n",a,b,c);

      return 0;
}
```

Laddered if-else

Γενικά

```
if (mycase == 1)
{
        print ;
}
else
{
        if (mycase == 2)
        {
        }
        else
        {
                if (mycase == 3)
                {
                }
                else
                {
                        if (mycase == 4)
                        {
                        }
                        else
                        {
                                /* nothing of the above */
                                /* ok to solve*/
                        }
                }
        }
}
```

Για το τριώνυμο

```
if  ( (a == 0) && (b == 0) && (c == 0))
{
        printf("Always holds!\n");
}
else
{
        if (a == 0)
        {
                /* solve & print accordingly */
        }
        else
        {
                /* still have to check */
                /* if b*b-2*a*c is positive */

                [...];

                if (D < 0)
                {
                }
                else
                {
                }
        }
}
```

Logic AND: if ((a == 0) || (b == 0) || (c = 0))

Logic OR   : if ((a == 0) && (b == 0) && (c = 0))

Δήλωση συναρτήσεων και εμβέλεια (scope) μεταβλητών

result = sqrt(D);

output = sqrt(input);

output = foo(input1, input2, ...);

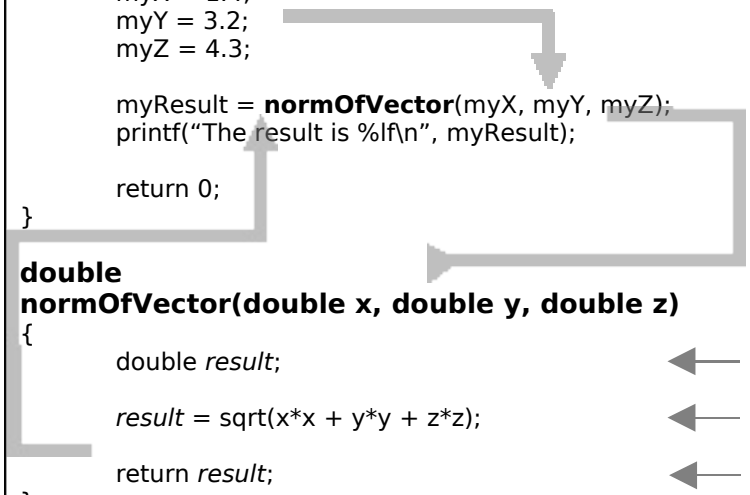
Παράδειγμα predefinition:

double normOfVector(double x, double y, double z);

Note: int, float, double compatibility

Example: normOfVector()

```
#include <stdio.h>

double normOfVector(double x, double y, double z);

int
main()
{
        double myResult;
        double myX, myY, myZ;

        myX = 1.4;
        myY = 3.2;
        myZ = 4.3;

        myResult = normOfVector(myX, myY, myZ);
        printf("The result is %lf\n", myResult);

        return 0;
}

double
normOfVector(double x, double y, double z)
{
        double result;

        result = sqrt(x*x + y*y + z*z);

        return result;
}
```



```
#include <stdio.h>

double normOfVector(double, double, double);

int
main()
{
        double myResult;
        double x, y, z;

        x = 1.4;
        y = 3.2;
        z = 4.3;

        myResult = normOfVector(x, y, z);
        printf("The result is %lf\n", myResult);

        return 0;
}

double
normOfVector(double x, double y, double z)
{
        double myResult;

        myResult = sqrt(x*x + y*y + z*z);

        return myResult;
}
```

Example: normOfVector()

```
#include <stdio.h>

double normOfVector(double, double, double);

int
main()
{
        double myResult;
        double x, y, z;

        x = 1.4;
        y = 3.2;
        z = 4.3;

        myResult = normOfVector(x, y, z);
        printf("The result is %lf\n", myResult);

        return 0;
}

double
normOfVector(double x, double y, double z)
{
        double myResult;

        myResult = sqrt(x*x + y*y + z*z);

        return myResult;
}
```

- Identify function
  - double
  - normOfVector(double, double, double);
- Allocate variables myResult, x, y, z
- Assign values to myX,myY,myZ
- Call normOfVector(myX,myY,myZ)
  - tmp: [1.4][3.2][4.3]
  - normOfVector(1.4, 3.2, 4.3);
  - myResult = [5.539856] normOfVector
- myResult = [5.539856]
- forget tmp

Initial state
[myResult, x, y, z.....]
[[?][1.4][3.2][4.3].....]

Function call
[myResult, x, y, z.....[double 0][double 1][double 2]]
[[?][1.4][3.2][4.3].....[1.4][3.2][4.3]]

Return
[myResult, x, y, z.....[**5.54**][1.4][3.2][4.3]]
[[**5.54**] [1.4][3.2][4.3]..... [**5.54**][1.4][3.2][4.3]]

Ready to print
[myResult, x, y, z, ...]
[[**5.54**] [1.4][3.2][4.3].....]

printf("The result is %lf\n", myResult);

Είσοδος από το πληκτρολόγιο: int scanf (char *format, ...);

```
#include <stdio.h>

int main ()
{
        int i;

        printf ("Enter an integer: ");
        scanf ("%d",&i);

        printf ("The integer you entered is: %d\n",i);

        return 0;
}
```

```
#include <stdio.h>

int
main()
{
        float myfloat;
        int err;

        printf ("Enter an float: ");
        err = scanf("%f", &myfloat);
        printf("The error code is %d\n",err);
        if (err == 0)
        {
                printf("I cannot understand this float\n");
        }
        else
        {
                printf ("The float you entered is: %f\n", myfloat);
        }

        return 0;
}
```

Γιατί ως παραμετρος εισόδου?

- **Έλεγχος / error-handling**
- **Πολλαπλή έξοδος**

Είσοδος	**Εκτύπωση**
err = scanf("%d", &myInt);	printf("int: %d, err: %d\n", myInt, err);
err = scanf("%f", &myFloat);	printf("float: %f, err: %d\n", myFloat, err);
err = scanf("%lf", &myDouble);	printf("double: %f, err: %d\n", myDouble, err);

Βρόχοι επανάληψης. Η περίπτωση του while

```
#include <stdio.h>

#define MY_LIMIT     10

int
main()
{
        int n;

        n = 0;

        while (n < MY_LIMIT)
        {
                printf ("n is now %d\n", n);
                n = n + 1;
        }
        printf("At last n is now %d\n",n);

        return 0;
}
```

Πως γίνεται για μεταβλητό n? scanf

```
int
main()
{
        float myVariable;
        int haveReadIt = 0;

        while (haveReadIt == 0)
        {
                int error;

                printf("Enter float:");
                error = scanf("%f",&myVariable);
                if (error == 0)  /* input was wrong*/
                {
                        printf("Wrong input try again.\n");
                        haveReadIt = 0;
                }
                else  /* input was OK */
                {
                        printf("OK. a is %f\n", myVariable);
                        haveReadIt = 1;
                }
                getchar();
        }
        return 0;
}
```

Πως στο τριώνυμο? Διάβάζουμε πρώτα μεταβλητές σωστά και μετά η διερεύνηση.

```c
#include <stdio.h>

int
main()
{
        int a,b,c, haveReadIt;

        haveReadIt = 0; /* for a */
        while (haveReadIt == 0)
        {
                int error;
                error = scanf("%d",&a);
                if (error == 0)
                {
                        haveReadIt = 0;
                }
                else
                {
                        printf("a is
                        %d\n",n);
                        haveReadIt = 1;
                }
        }

        haveReadIt = 0;  /* for b */
        while (haveReadIt == 0)
        {
                int error;
                error = scanf("%d";,&a);
                if (error == 0)
                [...]
        }
        [...]

        /* kai meta h diereynhsh
        */
        if (a==0) ...
        {
        }

        return 0;
}
```

a

b

```c
double
readDoubleFromKeyboard()
{
      double result;
      int haveReadIt = 0;

       while (haveReadIt == 0)
       {
              int c, error;

              printf("Please enter a real number:");
              error = scanf("%lf",&result);
              haveReadIt = error;
              while (  (c = getchar()) != '\n')
              {
                    if (! isdigit(c))
                    {
                          haveReadIt = 0;
                    }
              }

              if (haveReadIt == 0)
              {
                    printf("\nWrong input try again:\n");
              }
              else
              {
                    printf("OK the number is %lf\n", result);
              }
       }

       return result;
}
```

```c
#include <stdio.h>

double
readDoubleFromKeyboard();

int
main()
{
      double a,b,c;

      a =
readDoubleFromKeyboard();
      b =
readDoubleFromKeyboard();
      c =
readDoubleFromKeyboard();

      printf("a=%lf\tb=%lf\tc=
%lf\n",a,b,c);

      [...]

      return 0;
}
```