

Functions

- Functions used to break problem down into small, "bite-sized" pieces.
- Functions have an optional type of return value, a name, and optional arguments
- Functions return at most, ONE value
- Functions must be either "prototyped" or declared prior to use. Good programming practices requires all functions to be prototyped.

1

Functions

type of value returned

name of function

parameters of function (variables in)

```
double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}
```

Single value returned by function

Alternate way of writing above function

```
double hyp(double a, double b)
{
    return sqrt(a*a + b*b);
}
```

2

Functions - complete program

```
#include <stdio.h>
#include <math.h>
double hyp(double a, double b);
void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}
double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}
```

prototype (note semi-colon)

actual function definition
(NO semi-colon)

3

Functions - scope

```
#include <stdio.h>
#include <math.h>
double hyp(double a, double b);
void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}
double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}
```

x

y

h

4

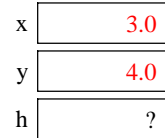
Functions - scope

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

double hyp(double a, double b);

void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}

double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}
```



5

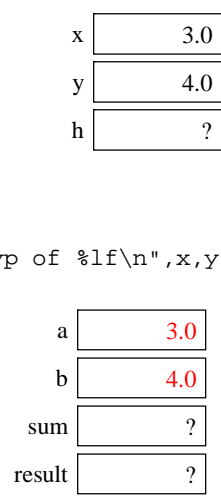
Functions - scope

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

double hyp(double a, double b);

void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}

double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}
```



6

Functions - scope

```

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

double hyp(double a, double b);

void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}

double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}

```

x	<input type="text" value="3.0"/>
y	<input type="text" value="4.0"/>
h	<input style="border: 1px solid black;" type="text" value="?"/>

a	<input type="text" value="3.0"/>
b	<input type="text" value="4.0"/>
sum	<input type="text" value="25.0"/>
result	<input style="border: 1px solid black;" type="text" value="?"/>

7

Functions - scope

```

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

double hyp(double a, double b);

void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}

double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}

```

x	<input type="text" value="3.0"/>
y	<input type="text" value="4.0"/>
h	<input style="border: 1px solid black;" type="text" value="?"/>

a	<input type="text" value="3.0"/>
b	<input type="text" value="4.0"/>
sum	<input type="text" value="25.0"/>
result	<input type="text" value="5.0"/>

8

Functions - scope

```

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

double hyp(double a, double b);

void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}

double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}
    
```

x	<input type="text" value="3.0"/>
y	<input type="text" value="4.0"/>
h	<input style="border: 1px solid black;" type="text" value="?"/>
a	<input type="text" value="3.0"/>
b	<input type="text" value="4.0"/>
sum	<input type="text" value="25.0"/>
result	<input type="text" value="5.0"/>

9

Functions - scope

```

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

double hyp(double a, double b);

void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
}

double hyp(double a, double b)
{
    double sum, result;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}
    
```

x	<input type="text" value="3.0"/>
y	<input type="text" value="4.0"/>
h	<input style="color: red;" type="text" value="5.0"/>

10

NOTE - a and b are NOT copied back to x and y

Exercise - What prints (if 5, 12 entered)

```

#include <stdio.h>
#include <math.h>
double hyp(double a, double b);
void main()
{
    double x,y,h;
    printf("Enter two legs of triangle: ");
    scanf("%lf %lf",&x,&y);
    h=hyp(x,y);
    printf("Trgle w legs %lf and %lf has hyp of %lf\n",x,y,h);
} //ANSWER: Trgle w legs 5.000000 and 12.000000 has hyp of 5.000000

double hyp(double a, double b)
{
    double sum, result;
    a = 3;
    b = 4;
    sum = a*a + b*b;
    result = sqrt(sum);
    return result;
}

```

x
 y
 h

 a
 b
 sum
 result

11

Functions - pass by address

```

#include <stdio.h>
#include <math.h>
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th);
void main()
{
    double x,y,h,r,th;
    printf("Enter x, y components of vector: ");
    scanf("%lf %lf",&x,&y);
    get_r_theta(x,y,&r,&th);
    printf("Vector with x=%lf and y=%lf
           has r=%lf, theta=%lf\n",x,y,r,th);
}

void get_r_theta(double a, double b,
                double *adr_r, double *adr_th)
{
    double sum;
    sum = pow(a,2)+pow(b,2); //or a*a+b*b;
    *adr_r = sqrt(sum);
    *adr_th = atan2(b,a);
}

```

x 4600
 y 4608
 r 4610
 th 4618

12

Functions - pass by address

```

#include <stdio.h>
#include <math.h>
void get_r_theta(double a, double b,
                 double *adr_r, double *adr_th);
void main()
{
    double x,y,h;
    printf("Enter x, y components of vector: ");
    scanf("%lf %lf",&x,&y); // user enters 3,4
    get_r_theta(x,y,&r,&th);
    printf("Vector with x=%lf and y=%lf
           has r=%lf, theta=%lf\n",x,y,r,th);
}

void get_r_theta(double a, double b,
                 double *adr_r, double *adr_th)
{
    double sum;
    sum = pow(a,2)+pow(b,2); //or a*a+b*b;
    *adr_r = sqrt(sum);
    *adr_th = atan2(b,a);
}

```

x	3.0	4600
y	4.0	4608
r	?	4610
th	?	4618

13

Functions - pass by address

```

#include <stdio.h>
#include <math.h>
void get_r_theta(double a, double b,
                 double *adr_r, double *adr_th);
void main()
{
    double x,y,h;
    printf("Enter x, y components of vector: ");
    scanf("%lf %lf",&x,&y);
    get_r_theta(x,y,&r,&th);
    printf("Vector with x=%lf and y=%lf
           has r=%lf, theta=%lf\n",x,y,r,th);
}

void get_r_theta(double a, double b,
                 double *adr_r, double *adr_th)
{
    double sum;
    sum = pow(a,2)+pow(b,2); //or a*a+b*b;
    *adr_r = sqrt(sum);
    *adr_th = atan2(b,a);
}

```

x	3.0	4600
y	4.0	4608
r	?	4610
th	?	4618
a	3.0	7380
b	4.0	7388
adr_r	4610	7390
adr_th	4618	7394
sum	?	7398

14

Functions - pass by address

```

#include <stdio.h>
#include <math.h>
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th);
void main()
{
    double x,y,h;
    printf("Enter x, y components of vector: ");
    scanf("%lf %lf",&x,&y);
    get_r_theta(x,y,&r,&th);
    printf("Vector with x=%lf and y=%lf
           has r=%lf, theta=%lf\n",x,y,r,th);
}
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th)
{
    double sum;
    sum = pow(a,2)+pow(b,2); //or a*a+b*b;
    *adr_r = sqrt(sum);
    *adr_th = atan2(b,a);
}

```

x	3.0	4600
y	4.0	4608
r	?	4610
th	?	4618
a	3.0	7380
b	4.0	7388
adr_r	4610	7390
adr_th	4618	7394
sum	25.0	7398

15

Functions - pass by address

```

#include <stdio.h>
#include <math.h>
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th);
void main()
{
    double x,y,h;
    printf("Enter x, y components of vector: ");
    scanf("%lf %lf",&x,&y);
    get_r_theta(x,y,&r,&th);
    printf("Vector with x=%lf and y=%lf
           has r=%lf, theta=%lf\n",x,y,r,th);
}
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th)
{
    double sum;
    sum = pow(a,2)+pow(b,2); //or a*a+b*b;
    *adr_r = sqrt(sum);
    *adr_th = atan2(b,a);
}

```

x	3.0	4600
y	4.0	4608
r	5.0	4610
th	?	4618
a	3.0	7380
b	4.0	7388
adr_r	4610	7390
adr_th	4618	7394
sum	25.0	7398

16

Functions - pass by address

```

#include <stdio.h>
#include <math.h>
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th);
void main()
{
    double x,y,h;
    printf("Enter x, y components of vector: ");
    scanf("%lf %lf",&x,&y);
    get_r_theta(x,y,&r,&th);
    printf("Vector with x=%lf and y=%lf
           has r=%lf, theta=%lf\n",x,y,r,th);
}
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th)
{
    double sum;
    sum = pow(a,2)+pow(b,2); //or a*a+b*b;
    *adr_r = sqrt(sum);
    *adr_th = atan2(b,a);
}

```

x	3.0	4600
y	4.0	4608
r	5.0	4610
th	36.87	4618
a	3.0	7380
b	4.0	7388
adr_r	4610	7390
adr_th	4618	7394
sum	25.0	7398

17

Functions - pass by address

```

#include <stdio.h>
#include <math.h>
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th);
void main()
{
    double x,y,h;
    printf("Enter x, y components of vector: ");
    scanf("%lf %lf",&x,&y);
    get_r_theta(x,y,&r,&th);
    printf("Vector with x=%lf and y=%lf
           has r=%lf, theta=%lf\n",x,y,r,th);
}
void get_r_theta(double a, double b,
                double *adr_r, double *adr_th)
{
    double sum;
    sum = pow(a,2)+pow(b,2); //or a*a+b*b;
    *adr_r = sqrt(sum);
    *adr_th = atan2(b,a);
}

```

x	3.0	4600
y	4.0	4608
r	5.0	4610
th	36.87	4618

18