

MA 511: Computer Programming

Lecture 14

http://www.iitg.ernet.in/psm/indexing_ma511/y08/index.html

Partha Sarathi Mandal

psm@iitg.ernet.ac.in

Dept. of Mathematics, IIT Guwahati

Semester 1, 2008-09

Mon 10:00-10:55 Tue 11:00-11:55 Fri 9:00-9:55 Class: 1G2

MA512 Lab : Wed 14:00-16:55

Passing function to other function

```
float guest1(float x, float y);  
float guest2(float x, float y);  
float host(float (*pt) (float x, float y));  
  
main(){  
    float x, y;  
    x = host(guest1);  
    printf("x = %f\n",x);  
    y = host(guest2);  
    printf("y = %f\n",y);  
} /* end of main */
```

```
float guest1(float x, float y){  
    return(x+y);  
}  
float guest2(float x, float y){  
    return(x*y);  
}  
  
float host(int (*pt) (float x, float y)){  
    float a=5.0, b=3.5, c;  
    c=(*pt)(a,b);  
    return(c);  
}
```

Dynamic memory allocation

```
int x[10];
```

Or

```
int *x;
```

```
x = (int *) malloc(10*sizeof(int));
```

- **malloc** library function reserves a block of memory size is equivalent to 10 integer for the above example.
- `stdlib.h` or `malloc.h` header file is required.
- **malloc** return a pointer to the beginning of the allocated space.
- in general **malloc(a)** allocate **a** bytes of memory.

Example

```
#include<stdio.h>
#include<stdlib.h>
main(){
    int *a, i, n;
    scanf("%d",&n);
    a = (int *) malloc(n*sizeof(int));
    for(i=0; i<n; i++){
        scanf("%d", a+i);
    }
    for(i=0; i<n; i++){
        printf("%d", *(a+i));
    }
}
```