Data Structures

lecture 8

6-10-2022

Last Session Quick Revision

Dereferencing a Pointer

- Finding out value at the address stored in the pointer
- int v = 10;
- int* ptr = &v;
- printf(“%d”,ptr);
- printf(“%d”,*ptr);

Pointer to Pointer

- int v = 10; //variable
- int* p1 = &v; //pointer to int variable
- int** p2 = &p1; //pointer to int* variable

Pointers and Arrays

```
void main() {
    int a[ ] = {1,2,3,4,5}, *p;
    p = a;
    ++*p;
    printf("%d ", *p);
   p += 2;
    printf("%d ", *p);
   }
```

Pointers to Structure

Pointers to Structure

```
struct complex
    {
        int real;
        int img
    };
    struct complex c1;    //variable of type complex
    struct complex* ptr = &c1; //pointer to structure
```

Dynamic Memory Allocation (Heap)

Dynamic Memory Allocation

- Refers to allocating memory on heap
- C uses special functions for it
  - malloc()
  - calloc()
- Heap Memory mgmt is not automatic:
  - if you allocate memory on heap manually (malloc/calloc)
  - You must de allocate it manually (free)

malloc

- Allocates the specified size of memory on heap.
- Starting address of allocated memory will be stored on stack in pointer
- ptr = (castType*) malloc(size in bytes);