Review C:

Stack

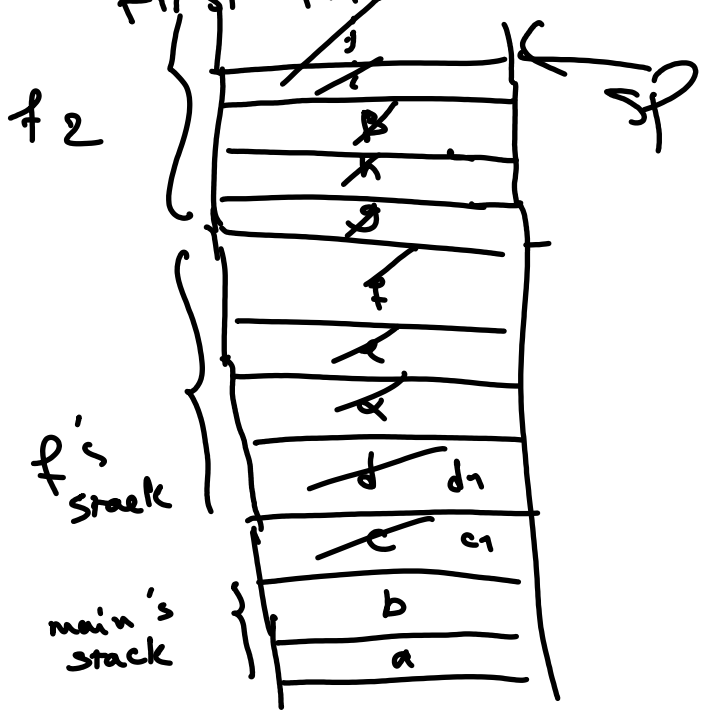
LIFO
Last In first Out

Queue

FIFO first In first Out

call stack

```
int main()
    int a, b;
    ≡
    f(c,d);        → return addr
    ≡ α
    ≡ a+4
{
int  f(int c1, int d1){
        int e, f;
        ≡
        f2(g,h);   → return addr
        ≡ β
}
```

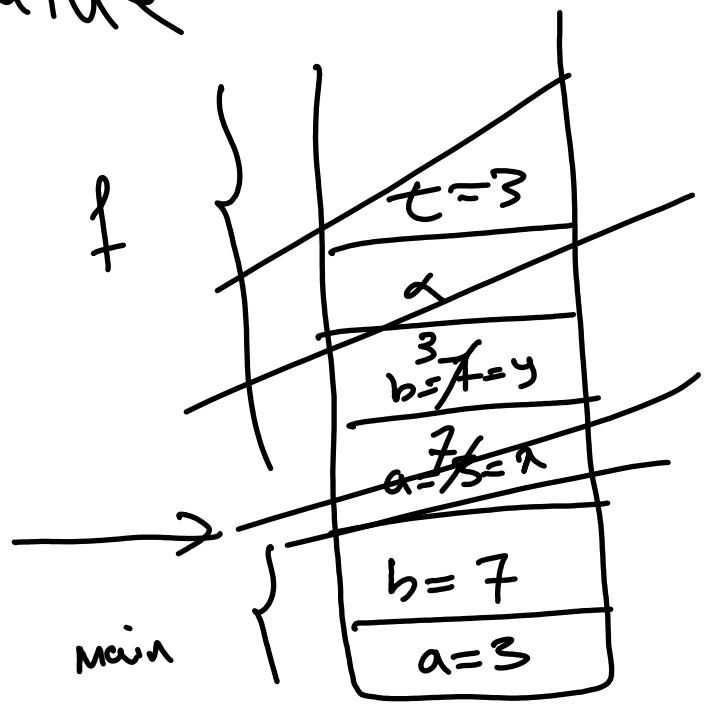



```
int  f2(int g1, int h1){
        int i, j;
        ≡
}
```

calling function { call by value
call by reference

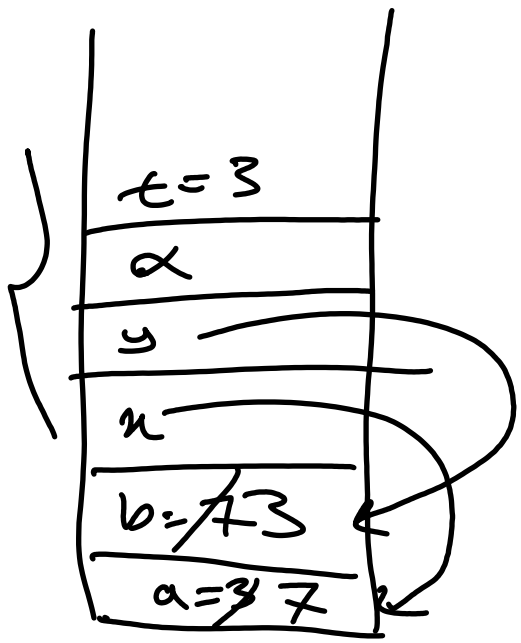
call by value

```
int main()
    int a=3, b=7;
    f(a,b);
}
int f(int x, int y){
    int t = x;
    x = y;
    y = t;
}
```



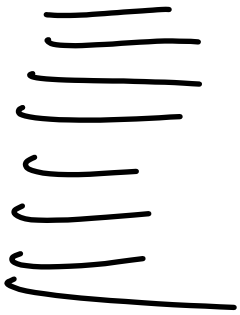

```
int f2(int *x, int *y)
    int t = *x;
    *x = *y;
    *y = t;
}
```

```
main(){
    int a=3, b=7;
    f2(&a, &b);
```



call by reference

struct student {



};

f (const struct student * st)

student * ptr1

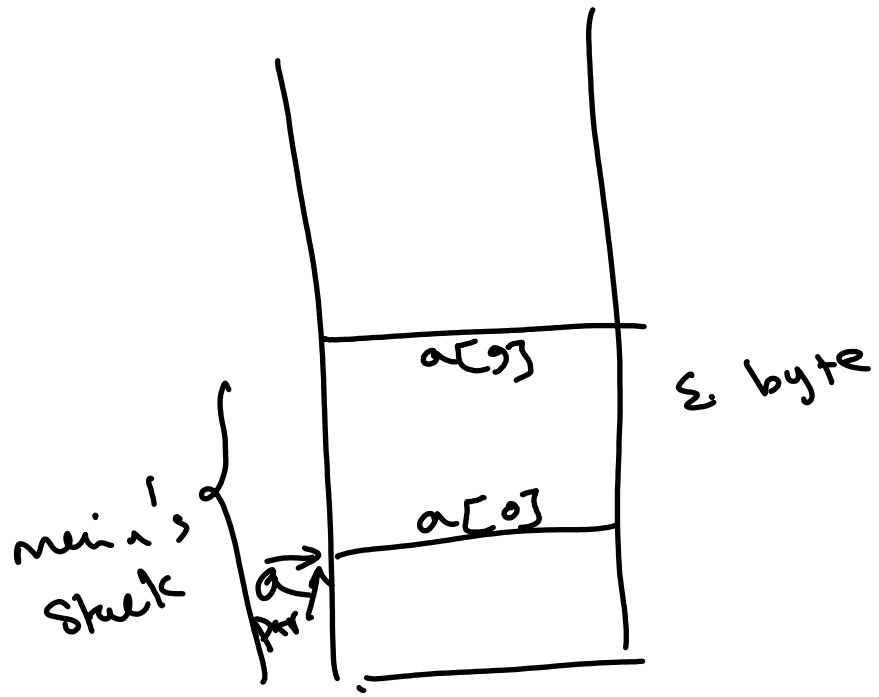
int * ptr2

sizeof (ptr1) = sizeof(ptr2) = 4 byte

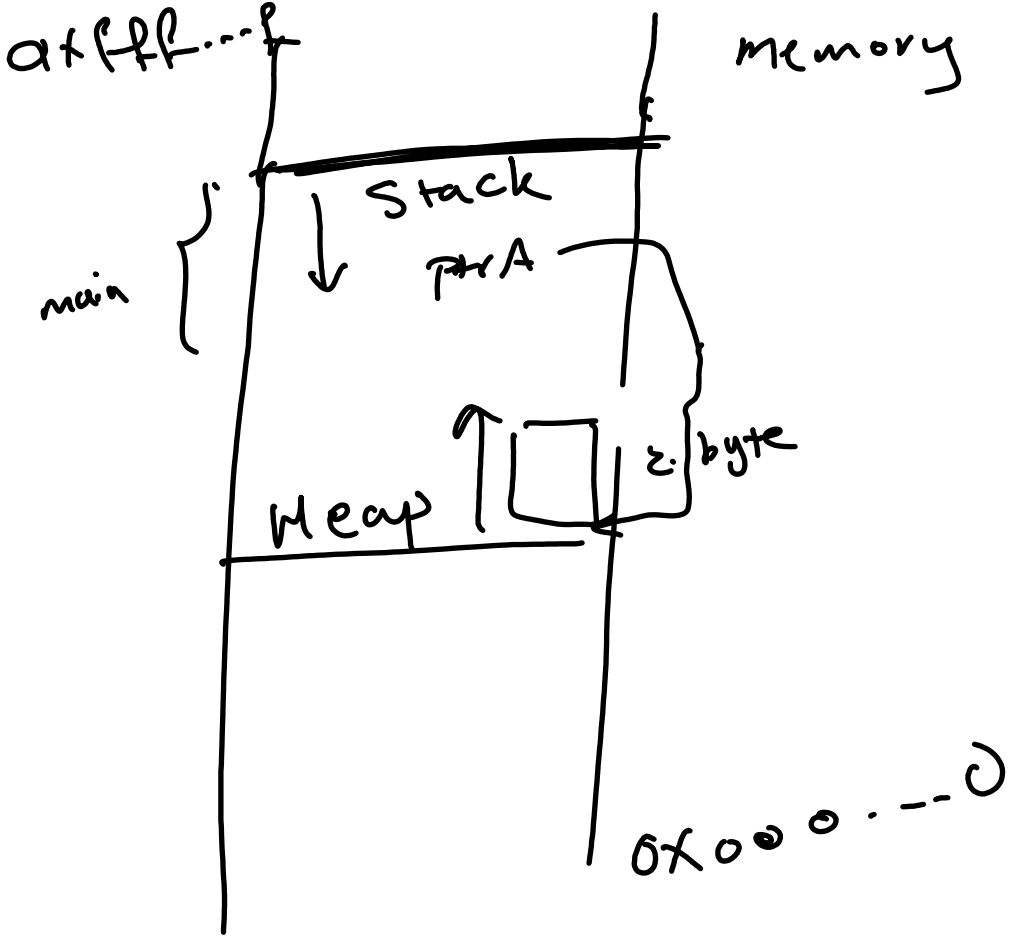
32 bit

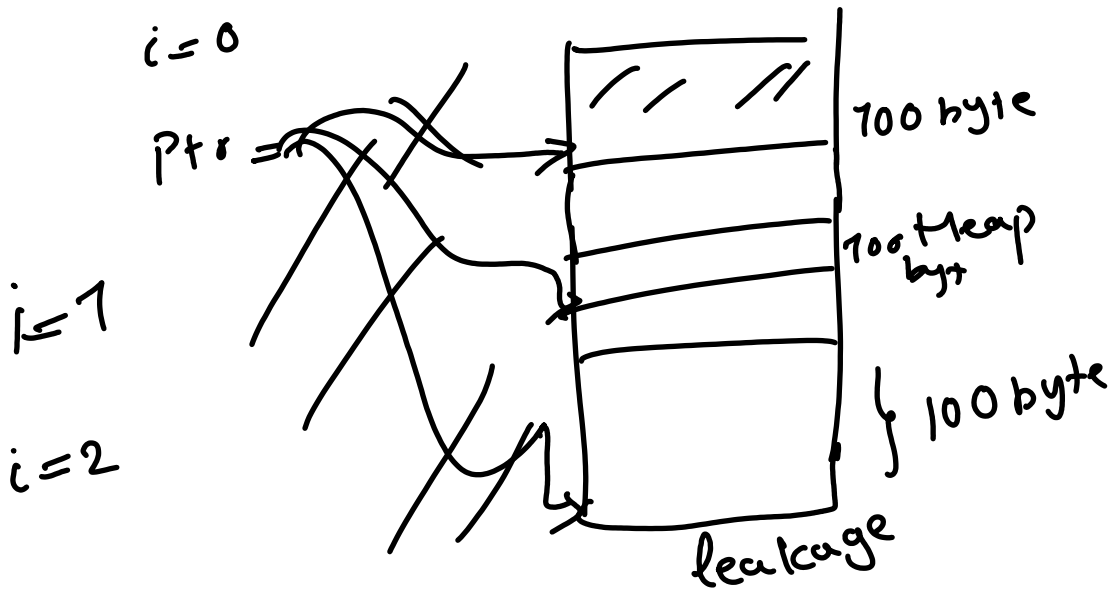
st->name.. -- = 8 byte

64 bit

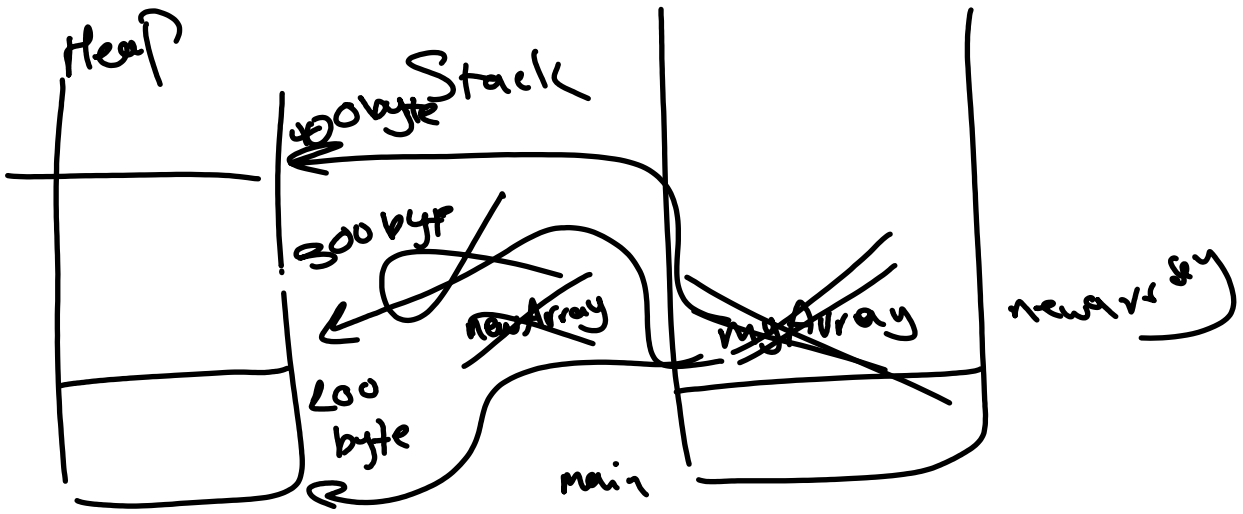


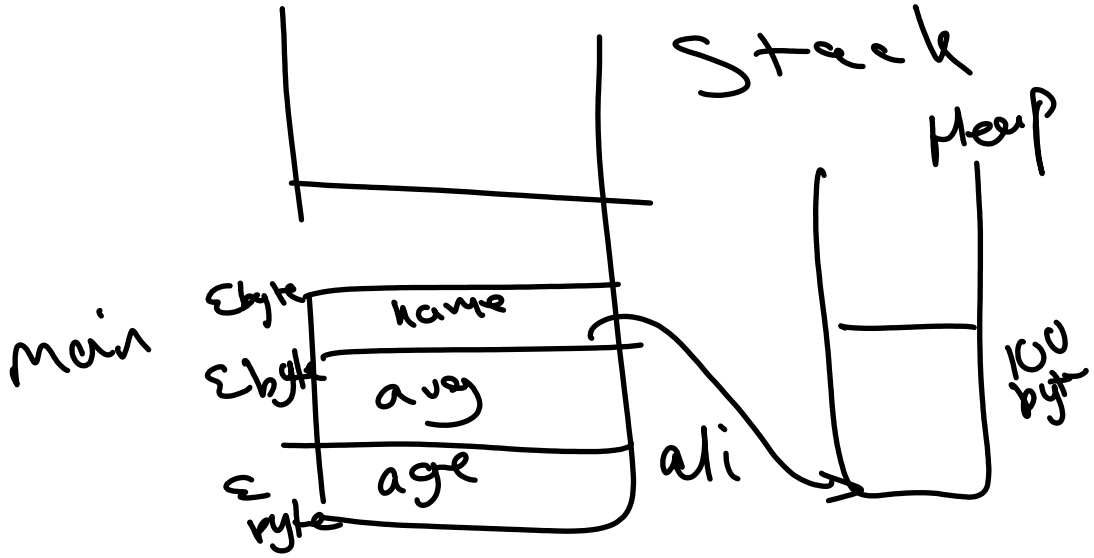
Heap




i=0
Ptr
i=1
i=2
100 byte
100 heap byte
100 byte
leakage

Sample 2


Heap
Stack
20 byte
300 byte
newArray
myArray
newArray
200 byte
main


Stack
Heap
main
Ebyte
name
Ebyte
avg
age
ali
100 byte