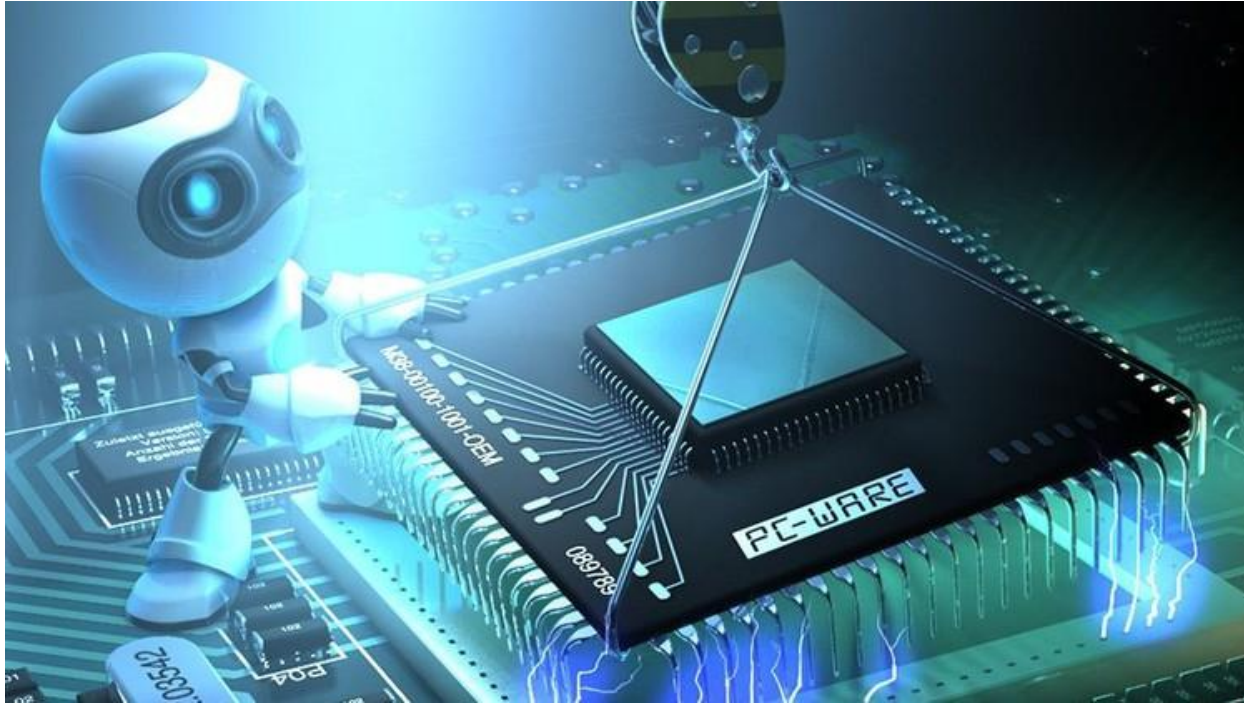
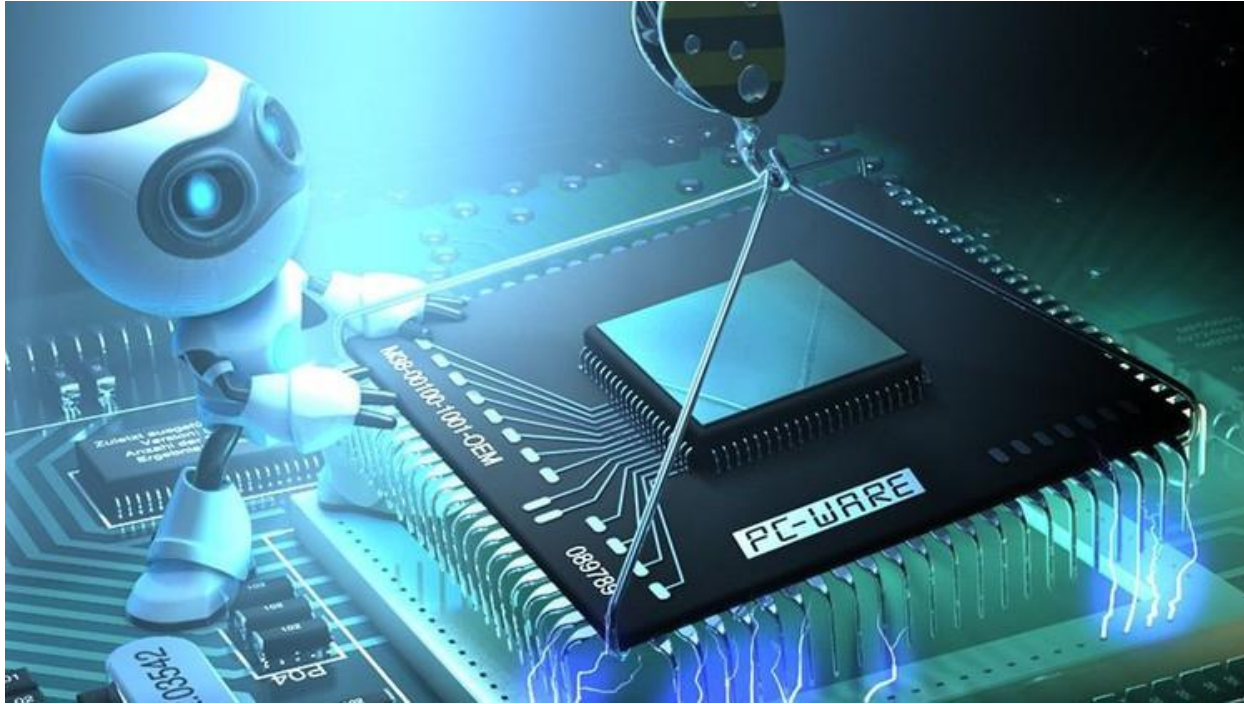


Laboratory Session 2



Practise - Problem 2



Practise - Problem 2



2. Translate the Find routine to x86 assembler.

```
int Find(S1 v[], S1 X, int N)
{
    int low, high, mid;
    int found = -1;

    low = 0;
    mid = low;
    high = N-1;
    while (low <= high) {
        found = FindElement(&low, &high, &mid, X, v);
        if (found >= 0) break; }
    return found;
}
```





Problem 2

2.

```
int Find(S1 v[], S1 X, int N)
{
    int low, high, mid;
    int found = -1;

    low = 0;
    mid = low;
    high = N-1;
    while (low <= high) {
        found = FindElement(&low, &high, &mid,
                           X, v);
        if (found >= 0) break;
    }
    return found;
}
```

Part 1/3

.text

.align 4

.globl Find

.type Find,@function

Find:

pushl %ebp

movl %esp, %ebp

movl \$-1, -4(%ebp)

found = -1

movl \$0, -16(%ebp)

low = 0

movl \$0, -8(%ebp)

mid = low

movl 32(%ebp), %ecx

%ecx = N

decl %ecx

%ecx = N-1

movl %ecx, -12(%ebp)

high = N-1

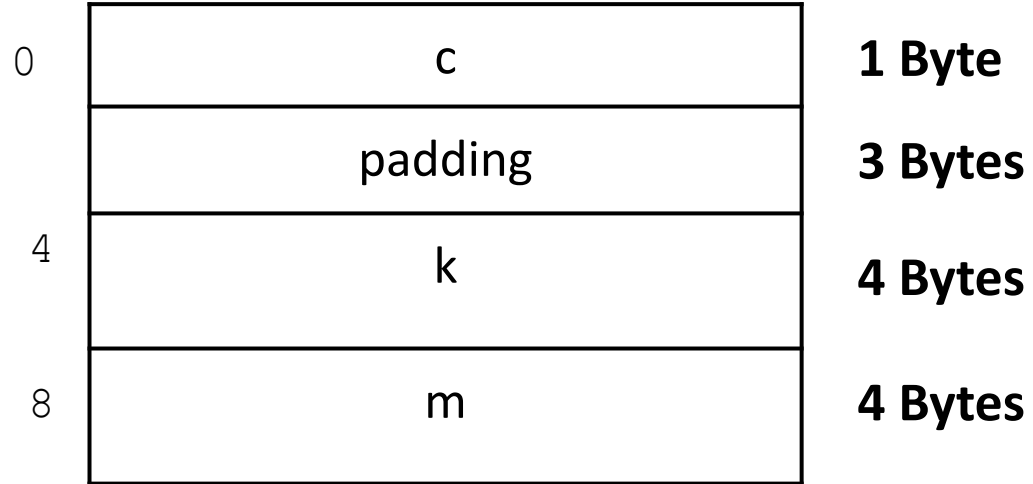
x86

Lucas Bazzilio - Udemv

Previous Study - Problem 1



```
typedef struct {  
    char c;  
    int k;  
    int *m;  
} S1;
```



Total size of struct S1: **12 bytes**



Problem 2

2.

```
int Find(S1 v[], S1 X, int N)
{
    int low, high, mid;
    int found = -1;

    low = 0;
    mid = low;
    high = N-1;
    while (low <= high) {
        found = FindElement(&low, &high, &mid,
        X, v);
        if (found >= 0) break; }
    return found;
}
```

Part 2/3

while:	movl -12(%ebp), %ecx	# high
	compl %ecx, -16(%ebp)	# compare high, low
	jg endwhile	
	# prepare stack to call FindElement	
	pushl 8(%ebp)	# &v
	pushl 28(%ebp)	# X.m
	pushl 24(%ebp)	# X.k
	pushb 20(%ebp)	# X.c
	leal -8(%ebp), %ecx	
	pushl %ecx	# &mid
	leal -12(%ebp), %ecx	
	pushl %ecx	# &high
	leal -16(%ebp), %ecx	
	pushl %ecx	# &low





Problem 2

2.

```
int Find(S1 v[], S1 X, int N)
{
    int low, high, mid;
    int found = -1;

    low = 0;
    mid = low;
    high = N-1;
    while (low <= high) {
        found = FindElement(&low, &high, &mid,
                           X, v);
        if (found >= 0) break; }
    return found;
}
```

Part 3/3

call

call FindElement

addl \$28, %ebp

found is already in %eax

compl \$0, %eax

jle endwhile

jmp while

endwhile:

ret

movl %ebp, %esp

popl %ebp

