

You may be asked to demonstrate/explain your work to the tutor, if you are absent/unavailable or fail to demonstrate properly, zero marks will be awarded.

IMPORTANT: Submission Format

Copy and paste the question and then write your answer. If it is a programming question copy and paste your code from text editor followed by the screenshots of the output window. Marks will be deducted if this format is not followed. You need to follow the exact sequential number as in the tut sheet.

Lab exercises

1. Give memory snapshots after each of these sets of statements is executed:

a.

```
int a=1, b=2, *ptr;  
...  
ptr = &b;
```

b.

```
int a=1, b=2, *ptr=&b;  
...  
a = *ptr;
```

c.

```
int a=1, b=2, c=5, *ptr=&c;  
...  
b = *ptr;  
*ptr = a;
```

d.

```
int a=1, b=2, c=5, *ptr;  
...  
ptr = &c;  
c = b;  
a = *ptr;
```

e.

```
double x=15.6, y=10.2, *ptr_1=&y, *ptr_2=&x;
...
*ptr_1 = *ptr_2 + x;
```

f.

```
int w=10, x=2, *ptr_2=&x;
...
*ptr_2 -= w;
```

2. Assume that an array `g` is defined with the following statement:

```
int g[]={2,4,5,8,10,32,78};
int *ptr1=&g[0], *ptr2=&g[3];
```

Give a **diagram** of the memory allocation, including the array values. Also indicate the offset values from the initial value in the array. Using this information, give the value of the following references:

- a. `*g`
- b. `*(g+1)`
- c. `*g+1`
- d. `*(g+5)`
- e. `*ptr1`
- f. `*ptr2`
- g. `*(ptr1+1)`
- h. `*(ptr2+2)`

3. Assume that an integer array `x` is defined by the following statements:

```
int x[2][4]={{1,8,7,6},{2,4,-1,0}}, *xptr=&x[0][0];
```

Draw a memory allocation **diagram**, and give the value indicated by each of the following references:

- a. `*xptr`
- b. `*(xptr+2)`
- c. `*xptr + 2`
- d. `*(xptr+1) + *(xptr+3)`

4. A text file contains grade details of five students. This file formatted like name followed by marks for four subjects. Contents of a sample text file is given below:

```
Peter
55
66
44
67
Lilly
100
```

90
43
89
John
34
56
78
65
Mary
45
56
78
90
Alex
30
45
65
54

Write a c program that reads the contents of the file and display it on the screen. Use the following structure for this program.

```
struct student
{
    char name[10];
    int marks[4];
};
```