

YILDIZ TECHNICAL UNIVERSITY
FACULTY OF ELECTRICAL AND ELECTRONICS ENGINEERING / DEPARTMENT OF CONTROL
AND AUTOMATION ENGINEERING

Name Surname:	Student No:	Signature:
Programming Languages	06.12.2019 16:00	Duration: 80 min
Midterm 2		
Asst. Prof. H. İrem TÜRKMEN		

1. Write the outputs of the given programs.

```
#include <stdio.h>
int x=5;
int func(int a, int *b)
{
    int c;
    a--;
    *b=*b+2;
    c=a+3;
    x++;
    printf("a:%d b:%d c:%d x:%d\n",a,*b,c,x);
    return c;
}
int main()
{
    int a=4,b=5,c;
    c=func(a,&b);
    x++;
    printf("a:%d b:%d c:%d x:%d\n",a,b,c,x);
    return 0;
}
```

```
a:3 b:7 c:6 x:6
a:4 b:7 c:6 x:7
```

```
#include <stdio.h>
int main()
{
    int arr[5] = {1, 2, 3, 4, 5};
    int *p,j;
    p=&arr[1];
    fun(4, p);
    for(j=0; j<5; j++)
        printf("%d ",arr[j]);
    return 0;
}
void fun(int n, int *ptr)
{
    int i=0;
    while(i < n-1)
    {
        *(ptr+i) = *(ptr+i+1);
        i++;
    }
}
```

```
1 3 4 5 5
```

2. Please choose the correct options. More than one selection is permitted from answer options.

2.1 Choose the correct combination of the following statements:

```
char *p; p = (char*) malloc(100);
```

- a. char p = *malloc(100);
b. char *p = (char) malloc(100);
c. char *p = (char*)malloc(100);
d. char *p = malloc(100);

2.2 Which are the correct usage of conditional operators in C?

- a. return (a>b)?(a:b)
b. a>b ? c=30
c. max = a>b ? a+3;--a;
d. printf("%d", a==b ? a: ++b);

2.3 Which of the followings are unary operators (takes a single operand)?

- a. |
b. sizeof
c. &&
d. ++

2.4 Which one of the given options are correct?

```
int i, j, k;
i = 3;
j =2*(i++);
k =2*(++i);
a. j = 6, k = 10    c. j = 6, k = 8
b. i = 5, k = 6    d. i = 4, j = 6
```

3. Height and name of a building is stored in the given structure below. Complete the program and write the following functions. Do not use global variables. Do not declare additional variables in main function.

- a) AllocateSpace() : Dynamically allocates space for an building array whose each element is a building.
- b) GetInfo() : Get the information of the buildings from the user and store in building array.
- c) ScanBuilds() : Scans building array and prints the name of the buildings which are taller than right and left neighbors.
(Ignore first and last buildings during scanning)

```

#include<stdio.h>
#include<stdlib.h>
typedef struct
{
    int height;      char nameofBuilding[20];
}BUILDING;
BUILDING* AllocateSpace(int);
void GetInfo(BUILDING *,int);
void ScanBuilds(BUILDING *,int);
int main()
{
    int i,n;
    BUILDING *buildingArray;
    printf("give the number of the buildings\n");
    scanf("%d",&n);
    buildingArray=AllocateSpace(n);
    GetInfo(buildingArray,n);
    ScanBuilds(buildingArray,n);
    return 0;
}

BUILDING* AllocateSpace(int n)
{

    return (BUILDING*)calloc(n,sizeof(BUILDING));
}

void GetInfo(BUILDING* array,int n)
{
    int i;
    for(i=0;i<n;i++)
        scanf("%d %s",&array[i].height,array[i].nameofBuilding);
}

void ScanBuilds(BUILDING* buildingArray,int n)
{
    int i;
    for (i=1;i<n;i++)
        if (((buildingArray[i].height)>(buildingArray[i+1].height)) && ((buildingArray[i].height)>(buildingArray[i-1].height)))
            printf("%s\n",buildingArray[i].nameofBuilding);
}

```