

Each question in this exam is worth 5 points. There are 20 questions.

### Embedded C

1) What is the difference between = and == ?

2) Using the following code skeleton, complete the function that sorts an array of 5 elements by reference.

```
void sort_array(int array[], int nelements)
{
    // ...
}
```

3. Consider the structure below. How can you assign the value 10 to the member i of the structure xxx using \*z?

```
#include<stdio.h>

struct myStructure
{
    int i;
    char *p;
};

int main()
{
    struct myStructure xxx;
    struct myStructure *z=&xxx;

    //... something goes here ...
}
```

4. How do you de-reference the void pointer (`pVoid`) to extract its associated value? Print its associated value.

```
#include<stdio.h>

int main()
{
    float floatValue=55.5;

    void *pVoid; //declaring a void pointer

    pVoid = &floatValue; // pVoid points to the address of intValue

    //... one line is missing here ...

    //... next line prints the floatValue through the
    //de-referenced void pointer ...

}
```

5. What is a function pointer? Write an example.

6. Write some code that will place the integer value 0X0034 into the absolute memory address 0xA3B5.

7. Write a SUM macro which takes two arguments and returns the sum of the two.

8. What is a register variable and why do we need them?

9. Write an example in which you modify the contents of a const int variable.

10. Using the following structure, write a function that will count the number of elements of a linked list.

```
#include <stdio.h>
#include <stdlib.h>

struct Node
{
    int Data;
    struct Node *Next;
}*Head;

// You must complete this function that counts number of elements
int length()
{

    return(count);
}

int main(int argc, char *argv[])
{
    int i=0;

    //Set HEAD as NULL
    Head=NULL;
    addBeg(num); //this function has been declared somewhere else
    printf("Number of elements=%d\n",length());
};
```

## **RTOS**

11. What are some of the common features offered in a commercial RTOS?
12. Each task has its own private context. What does this mean?
13. In a RTOS, what is the state of a task that is waiting for an event to occur?
14. Interrupt latency, footprint and context switching time are perhaps the most important factors considered when a RTOS is being selected. Could you briefly describe them?
15. What happens if the programmer disables all maskable interrupts and forgets to re-enable it?
16. What is an atomic operation?
17. What is priority inversion?

18. What is a non-preemptive scheduling (a.k.a. cooperative multi-tasking) algorithm?

19. What is a reentrant function?

20. Why RTOS developers tend to avoid the standard calloc and malloc functions?