King Saud University

College of Computer and Information Sciences
Computer Science Department

Course Code:	CSC 215
Course Title:	Procedural Programming
Semester:	Semester 1 of year 36/37
Exercises Cover Sheet:	**Final Exam**

Duration: 120 minutes

Student Name:	
Student ID:	
Student Section No.	

Computer Science B.Sc. Program: NCAAA: Intended Learning Outcomes (ILO) Student Outcomes ABET: Program Learning Outcomes (PLO) Student outcomes		**Question No. Relevant Is Hyperlinked**	**Covering %**
NCAAA	**1. Knowledge (NCAAA)** **Suggested verbs (list, name, record, define, label, outline, state, describe, recall, memorize, reproduce, recognize, record, tell, write)**	Exercise 2	20%
	(i) Use current techniques, skills, and tools necessary for computing practices; *The students learn how to use Integrated Development Environment to compile and run C programs. Students also learn the differences between procedural and object oriented languages*	Exercise 2	20%
NCAAA	**2. Cognitive Skills (NCAAA)** **Suggested verbs (estimate, explain, summarize, write, compare, contrast, diagram, subdivide, differentiate, criticize, calculate, analyze, compose, develop, create, prepare, reconstruct, reorganize, summarize, explain, predict, justify, rate, evaluate, plan, design, measure, judge, justify, interpret, appraise)**	Exercises 1&3	50%
ABET	b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution; *Students learn how to manage memory using dynamic memory allocation based on problem requirement analysis*.	Exercise 3	30%
	c. An ability to design, implement and evaluate a computer-based system, process, component or program to meet desired goals. *Students write procedural C programs.*	Exercise 1	20%

EXERCISE 1 Answer the following questions **(/30pts)**

1- Declare a structure called **student** with the following elements: **name** as character array of size 20, studentID as an int, **birthDate** as a structure whose elements are the integers day, month and year. (10pts)

2- Create a pointer to the structure student (from question 1) called **csc215list** and dynamically allocate memory for 30 students. (7pts)

3- Write the code to de-allocate memory used by the array **csc215list** (from question 2). (3pts)

4- Write the function **tolower** from the ctype library. You may **NOT** use any other function inside this function. (10pts)

```
int tolower(int c){




}
```

EXERCISE 2 Write True/ False **(/20pts)**

C can be used as a system programming language	
C is a structured programming language	
In C, memory management is done automatically	
UNIX is written in C, so it is easier to interface with UNIX operating systems if you write in C.	
C provides language constructs that map efficiently to machine instructions which makes C the best language choice when writing any program.	
One of the most important characteristics of procedural programming is that it relies on procedures that operate on data - these are two separate concepts. In object-oriented programming, these two concepts are bundled into objects.	
C has been written in assembly language	
In C, low-level access to computer memory is possible by converting machine addresses to typed pointers.	
C includes all features found in newer, more modern high-level languages	
C is closer to assembly language, so you can have finer control over what the computer is doing, and thereby make faster programs.	

EXERCISE 3 Select the correct answer **(/50pts)**

1. What is the correct value returned to the operating system upon successful completion of a program?

a) 0
b) 1
c) -1
d) Program do not return a value.

2. A declaration "short int" is used for variables

a) Which have a short duration in a program
b) Which have short names
c) Which may require less storage than normal integer
d) All of the above.

3. What will be the value of count after the following program is executed?

```
void main(){
   int count, digit=0;
   count = 1;
   while(digit <= 9)
   {
      printf("%d\n",++count);
      ++digit;
   }
}
```

a) 10
b) 9
c) 12
d) 11

4. The following lines, if included in a program will cause one of the following errors. Indicate the correct one

```
double c;
 scanf("%c",c);
```

a) runtime error
b) compile time error
c) typedef error
d) no error

5. Which of the following cannot be checked in a switch-case statement?

a) Character
b) Integer
c) Float
d) Enum

6. Consider the following program fragment: The correct values of a and b are

```
int main()
{
   int a,b,c;
   b = 3;
   a = 2;
   a = 2* (b++);
   c = 2* (++b);
   return 0;
}
```

a) a = 6, c = 8
b) a = 6, b = 3
c) b = 3, c = 6
d) a = 6, c = 10

7. How many times the string is printed?

```
#include<stdio.h>
int main()
{
   int x;
   for(x = -1; x<=10; x++)
   {
      if(x < 5)
         continue;
      else
         break;
      printf("Welcome\n");
   }
   return 0;
}
```

a) Infinite Times
b) 11
c) 0
d) 10

8. Point out the error, if any in the for loop.

```
#include<stdio.h>
int main()
{
   int i=1;
   for(;;)
   {
      printf("%d\n", i++);
      if(i>10)
        break;
   }
   return 0;
}
```

a) There should be a condition in the for loop
b) The two semicolons should be dropped
c) The for loop should be replaced with while loop.
d) No error

9. What will be the output of the program?

```
#include<stdio.h>
int main()
{
   int k, num = 30;
   k = (num < 10) ? 100 : 200;
   printf("%d\n", num);
   return 0;
}
```

a) 200
b) 30
c) 100
d) 500

10. How can you write a[i][j][k][l] in equivalent pointer expression?

a) (((***(a+i)+j)+k)+l)
b) ((**(*(a+i)+j)+k)+l)
c) (*(*(*(a+i)+j)+k)+l)
d) *(*(*(*(a+i)+j)+k)+l)

11. Which of the following are correct syntaxes to send an array as a parameter to function:
a) func(&array);
b) func(array);
c) func(*array);
d) func(array[size]);

12. What is the output of this C code?

```
void main()
{
        int x = 0;
        int *ptr = &5;
        printf("%d\n", *ptr);
}
```

a) 5
b) Address of 5
c) Nothing
d) Compile time error

13. What is the output of this C code?

```
int x = 0;
void main()
{
        int *ptr = &x;
        printf("%d\n", ptr);
        x++;
        printf("%d\n ", ptr);
}
```

a) Same address
b) Different address
c) Compile time error
d) Varies

14. What is the output of this C code?

```
void foo(int*);
int main()
{
        int i = 10, *p = &i;
        foo(p++);
}
void foo(int *p)
{
        printf("%d\n", *p);
}
```

a) 10
b) Some garbage value
c) Compile time error
d) Segmentation fault

15. What is the output of this C code?

```
int main()
{
        int i = 97, *p = &i;
        foo(&i);
        printf("%d ", *p);
}
void foo(int *p)
{
        int j = 2;
        p = &j;
        printf("%d ", *p);
}
```

a) A. 2 97
b) B. 2 2
c) C. Compile time error
d) D. Segmentation fault/code crash

16. Which of the following cannot be a structure member?

a) Another structure
b) Function
c) Array
d) None of the mentioned

17. If b is a pointer to a structure, which of the following accesses its member variable var.

a) b->var;
b) b.var;
c) b-var;
d) b>var;

18. What would be the output of the following program?

```
main()
{
        char str1[] = "Hello";
        char str2[] = "Hello";
        if(str1 == str2)
            printf("Equal");
        else
           printf("\nUnequal");
}
```

a) Error.
b) Equal.
c) Unequal.
d) None.

19. If the two strings are identical, then strcmp() function returns? +

a) 0
b) 1
c) -1
d) None

20. The function **strrch** in the string.h library returns

a) a pointer to the last occurrence of a character in a string
b) a pointer to the first occurrence of a character in a string
c) a pointer to the last occurrence of a string in a string
d) a pointer to the first occurrence of a string in a string

21. The function **islower** can be found in which library

a) string.h
b) type.h
c) ctype.h
d) stdlib.h

22. Which of the following differences between malloc and calloc is not true?

a) malloc allocates number of bytes passed as argument
b) calloc allocates the product of number of elements and the size of each element, which are both passed as arguments.
c) both malloc and calloc return void*
d) both malloc and calloc initialize allocated memory to all 0

23. Which of the following is NOT an error when using free:

a) Use free on a NULL pointer
b) Use free on a pointer that has already been freed
c) Use free on a memory address directly returned by malloc.
d) Use free on a memory address that has been statically allocated

24. Which of the following is not true about stderr?

a) It is an output stream for errors
b) It is assigned to a program just like stdin and stdout
c) It will cause the program, to exit
d) It will make the output appears on screen even if stdout is redirected

25. The following function implements which searching or sorting algorithm

```
int algorithm (int a[], int n, int key)
{
    int i;
    for(i = 0; i < n; i++)
    {
        if (a[i] == key)
            return i;
    }
    return -1;
}
```

a) Bubble sort
b) Selection sort
c) Linear search
d) Binary search