Quiz 1 *(first 10 questions are irrelevant to the topic)*

Question 11 **7.5 pts**

What is a programming paradigm?

1- A common set of hardware instructions for a specific kind of hardware.
2- A set of principles, concepts, and methods that is commonly accepted by members of a group or community.
3- A programming language is a paradigm.
4- A mathematical model.

Answer: 2

Question 12 **7.5 pts**

The compiler executes the program. (True/False)

Answer: False

Question 13 **7.5 pts**

Prolog is a functional programming language. (True/False)

Answer: False

Question 14 **7.5 pts**

Features of the logic paradigm include:

1- Expressing computations in terms of logic predicates.
2- Expressing computations in terms of Boolean expressions.
3- Using lambda calculus.
4- Classes and objects.

Answer: 1,4

Question 15 **7.5 pts**

Features of the functional paradigm include:

1- Expresses computation in terms of mathematical functions.
2- Expresses computations in terms of logic predicates.
3- Polymorphism.
4- Simpler semantics.

Answer: 1,4

Question 16 **7.5 pts**

Interpretation of a program is the direct execution of one statement at a time sequentially. (True/False) ***Answer:*** True

Question 17 **7.5 pts**

Features of imperative or procedural paradigm include:

1- Classes and objects.
2- Manipulation of named data (variables).
3- Conditional statements.
4- Encapsulation.

Answer: 2,3

Question 18 **7.5 pts**

Features of the object-oriented paradigm include:

1- Logic predicates.
2- Classes and objects.
3- Lambda calculus.
4- Inheritance.

Answer: 2,4

Question 19 **7.5 pts**

A programming language can belong to multiple paradigms. (True/False)

Answer: True

Question 20 **7.5 pts**

Logic programming languages divide the program into reasonable sized pieces named functions or modules or subroutines. (True/False)

Answer: False

Quiz 2

Question 1 **10 pts**

What is a data type?

1- A specialized format for organizing and storing data.
2- A storage location paired with an associated symbolic name.
3- A piece of information.
4- A set of primary values and the operations defined on these values.

Answer: 4

Question 2 **10 pts**

Interpretation of a program is the direct execution of one statement at a time sequentially. (True/False)

Answer: True

Question 3 **10 pts**

Thy syntactic structure of imperative programming languages normally include which of the following units.

1- Operators.
2- Keywords.
3- Identifiers.
4- Conditional statements.
5- Loop statements.
6- Variable declaration.

Answer: 4,5,6

Question 4 **10 pts**

What kind of error is in the following line:

```
int a = ((2 * 45) * (6 / 2) hello (4 + 90));
```

1- Lexical error.
2- Semantic error.
3- Syntactic error.
4- None of the above.

Answer: 3

Question 5 **10 pts**

The compiler executes the program. (True/False)

Answer: False

Question 6 **10 pts**

Compilation of a program is to execute all the statements of the program completely. (True/False)

Answer: False

Question 7 **10 pts**

The lexical structure of all programming languages is similar and normally includes which of the following units?

1- Identifiers.
2- Loop statements.
3- Keywords.
4- Operators.
5- Literals.
6- Variable declaration.

Answer: 1,3,4,5

Question 8 **10 pts**

During compilation, all statements of a program in a high-level language are converted (translated) to a low-level language (such as assembly language). (True/False)

Answer: True

Question 9 **10 pts**

Functional programming languages are low-level languages. (True/False)

Answer: False

Question 10 **10 pts**

The semantic structure of imperative programming languages normally include which of the following validations?

1- Type matching.
2- Division by zero.
3- A variable name should start with a letter '$' or '_'.
4- Parameter types in a function declaration should match these in the function call.
5- Statement should end with a ';'.
6- Unicity.

Answer: 1,4,6

Quiz 3

Question 1

What is the output of the following code?

```
int foo(int *n)
{
    *n = 30;
}

int main()
{
    int i = 15;
    foo(&i);
    printf("i=%d\n", i);
    i = 10;
    foo(&i);
    printf("i=%d\n", i);
    return 0;
}
```

1- i=15
   i=10

2- i=30
   i=30

3- i=15
   i=30

4- i=30
   i=10

Answer: 2

Question 2

Which code in C is equivalent to this code in Java?

```
int x = 5;
float y = 10.3f;
System.out.println(“hello “ + x + “ bye “ + y);
```

1-
```
int x = 5;
float y = 10.3;
printf(“hello %d bye %f”, &x, &y);
```

2-
```
int *x = 5;
float *y = 10.3;
printf(“hello %p bye %p”, x, y);
```

3-
```
int x = 5;
float y = 10.3f;
```

```
   printf(“hello %i bye %f”, x, y);
```

4-
```
int x = 5;
float y = 10.3;
printf(“hello %d bye %f”, x, y);
```

Answer: 4

Question 3

What does the below program print on the screen?

```
int main()
{
    int i = 3, *j, k;
    j = &i;
    printf(“%d\n”, i * *j * i - *j);
    return 0;
}
```

1- 78
2- 24
3- 60
4- 54

Answer: 2

Question 4

The following code is correct and prints "`Hello`". (True/False)

```
if (2 + 2 + 2 + 2)
    if (1)
        printf("Hello");
```

Answer: True

Question 5

Which of the following are NOT primitive data types in C?

1- `int`
2- `long`
3- `bool`
4- `char`
5- `double`
6- `short`
7- `float`
8- `string`

Answer:

`bool, string`

Question 6

What is the output of the following program?

```
typedef enum week { Mon, Tue, Wed, Thur, Fri, Sat, Sun } WEEK;
int main()
{
    WEEK day;
    day = Wed;
    printf("%d", day);
    return 0;
}
```

1- 2
2- 3
3- 4
4- 0

Answer: 2

Question 7

Which of the following lines prints "-15" on the screen?

```
int main()
{
    int i = 15;
    foo(&i);
    printf("i=%d\n", i);
    i = 10;
    foo(&i);
    printf("i=%d\n", i);
    return 0;
}
```

1-
```
int x = -15;
int *point = &x;
printf(“%d”, *point);
```

2-
```
int x = -15;
int *point = &x;
printf(“%d”, &x);
```

3-
```
int x = -15;
int *point = &x;
printf(“%p”, point);
```

4-
```
int x = -15;
printf(“%p”, &x);
```

Answer:

Question 8

What is the output of the below code?

```
int fun(int n) {
    if (n == 4)
        return n;
    else
        return 2 * fun(n + 1);

}

int main()
{
    printf("%d", fun(3));
    return 0;
}
```

1- 4
2- 8
3- 16
4- 32

Answer: 2

Question 9

Given the following struct:

```
struct contact
{
    char name[32];
    int phone;
    char email[32];
}
```

Which code can be used to create a contact and store data?

1-
```
struct contact x;
scanf(“%s”, &x.name);
scanf(“%d”, &x.phone);
scanf(“%s”, &x.email);
```

2-
```
struct contact x;
scanf(“%s”, &x.name);
scanf(“%d”, x.phone);
scanf(“%s”, &x.email);
```

3-
```
struct contact x;
scanf(“%s”, x.name);
scanf(“%d”, x.phone);
scanf(“%s”, x.email);
```

4-
```
struct contact x;
```

```
scanf(“%s”, x.name);
scanf(“%d”, &x.phone);
scanf(“%s”, x.email);
```

Answer: 2

Question 10

What is the output of the below code?

```
int foo(int *n) {
    *n = 30;
}

int main()
{
    int a[5] = {3, 1, 5, 20, 25};
    int i, j, m;
    i = *(a + 1) - 1;
    j = a[1] + 1;
    m = a[j] + a[i];
    printf("%d,%d,%d", i, j, m);
    return 0;
}
```

1- 0, 1, 6
2- 0, 2, 8
3- 0, 2, 5
4- 1, 2, 6

Answer: 2

Question 11

What is the output of the below code?

```
int i = 10;

int bar(int m, int* n)
{
    printf("i=%d k=%d l=%d\n", i, m, *n);
}

int foo(int k, int* l)
{
    printf("i=%d k=%d l=%d\n", i, k, *l);
    k = 3;
    *l = 4;
    bar(k, l);
}

int main()
```

```
{
    int j = 15;
    foo2(j, &i);
    printf("i=%d j=%d\n", i, j);
}
```

1- i = 10 k = 15 l = 10
   i = 10 k = 3 l = 4
   i = 10 j = 15

2- i = 10 k = 15 l = 10
   i = 10 k = 3 l = 10
   i = 10 j = 15

3- i = 4 k = 3 l = 4
   i = 4 j = 15

4- i = 10 k = 15 l = 10
   i = 4 k = 3 l = 4
   i = 4 j = 15

Answer: 4

Quiz 4

Question 1 — 10 pts

The **scope resolution operator** (: :) is used to overload a function or an operator in object-oriented paradigm. (True/False)

Answer: False

Question 2 — 10 pts

Which of the following options is the code in C++ for

a) A class Student that inherits from a class Person.

b) A constructor in Student that calls (is able to call) a constructor in Person

- When the body of the method is not relevant to answer the question, it has been replaced for a comment **// code**

1-

```
class Person {
  public:
    Person() { //code
    }
    Person(char* lName, int year) { // code
    }
  private:
    char* lastName;
    int yearOfBirth;
};

class Student : public Person {
  public:
    Student() {
      //code
    }
    Student(char* lName, int year, char* univer) :Person(lName, year) {
      //code
    }
  private:
    char *university;
};
```

2-

```
class Person {
  public:
    Person() {
      //code
    }
```

```
    Person(char* lName, int year) {
      // code
    }
  private:
    char* lastName;
    int yearOfBirth;
};

class Student extends Person {
  public:
    Student() {
      //code
    }
    Student(char* lName, int year, char* univer) {
      Person(lName, year);
      //code
    }
  private:
    char *university;
};
```

3-

```
class Person {
  public:
    Person() {
      //code
    }
    Person(char* lName, int year) : Student (char* lName, int year, char* univer) {
      // code
    }
  private:
    char* lastName;
    int yearOfBirth;
};

class Student : public Person {
public:
  Student() { //code
  }
  Student(char* lName, int year, char* univer) { //code
  }
private:
  char *university;
};
```

4-

```
class Person {
  public:
    Person() {
      //code
```

```
    }
    Person(char* lName, int year) {
      // code
    }
  private:
    char* lastName;
    int yearOfBirth;
};

class Student : public Person {
  public:
    Student() {
      //code
    }
    Student(char* lName, int year, char* univer) {
      Person(lName, year);
      //code
    }
  private:
    char *university;
};
```

Answer: 1

Question 3 **10 pts**

Given the following class definition:

```
class Rectangle {
  private:
    int width, height;
  public:
    void set_values (int,int);
    int area ();
};
```

Which of the following instruction(s) create an array of 2 Rectangles and initialize them (the 2 Rectangles) with values. Select ALL the possible options.

1-
```
Rectangle a[2];
a[0].set_values(1,1);
a[1].set_values(2,2);
Rectangle *a[2];
```

2-
```
a[0] = new Rectangle;
a[0]->set_values(1,1);
a[1] = new Rectangle;
a[1]->set_values(2,2);
```

3-
```
Rectangle *a = new Rectangle[2];
a[0]->set_values(1,1);
a[1]->set_values(2,2);
```

```
4- Rectangle a = new Rectangle[2];
   a[0].set_values(1,1);
   a[1].set_values(2,2);
```

Answer: 1,2

Question 4 **10 pts**

Which of the following classes creates and initializes correctly an static variable in C++?

```
1- class Something {
   public:
       static int v;
   };
   Something::v = 1;
2- class Something {
   public:
       static int v;
   };
   int Something::v = 1;
3- class Something {
   public:
       static int v;
   };
   int v = 1;
4- class Something {
   public:
       static int v;
   };
   v = 1;
```

Answer: 2

Question 5 **10 pts**

Running the following program.

How many times will the message "good bye!" be printed on the screen?

```
#include<iostream>
using namespace std;
class CSE {
  public:
    CSE(int v) {
      cout<<"constructor\n";
    }
    void add(int v) {
      cout<<"adding\n";
    }
```

```
    int remove(){
      cout<<"removing\n";
      return 0;
    }
    ~CSE() {
      cout<<"good bye!\n";
    }
};
int main(){
  CSE q1(5);
  CSE *q2 = new CSE(5);
  q1.add(2);
  q1.add(8);
  q1.remove();
  q2->remove();
  delete q2;
  return 0;
}
```

Answer: 2

Question 6 **10 pts**

In the following code, how many times the destructor of the class "**Base**" is executed?

```
#include<iostream>
using namespace std;
class Base {
  public:
    Base(int n) {
      cout<<"Base Constructor\n";
    }
    void function() {
      cout<<"function\n";
    }
    ~Base() {
      cout<<"Base destructor\n";
    }
};

class Derived : public Base {
  public:
    Derived(int n) : Base(n) {
      cout<<"Derived Constructor\n";
    }
    ~Derived() {
      cout<<"Derived destructor\n";
    }
};

int main(){
  Derived myPQ1(50);
  myPQ1.function();
```

```
  Derived *myPQ2 = new Derived(50);
  myPQ2->function();
  delete myPQ2;
  return 0;
}
```

Answer: 2

Question 7 **10 pts**

The principle behind the object-oriented paradigm consists of a number of programming concepts, which **does not** include the following:

1- Classes.
2- Polymorphism.
3- Pointers.
4- Arrays.
5- Inheritance.

Answer: 3,4

Question 8 **10 pts**

The following declaration allows all elements in the standard C++ library to be accessed in an unqualified manner (without the `std:: prefix`)

1- `#include <iostream.h>`
2- `using namespace iostream;`
3- `using namespace std;`
4- `#include <iostream>`

Answer: 3

Question 9 **10 pts**

In C++, implementations of member functions cannot be inside the class definition (for short functions) or outside of the class definition. (True/False)

Answer: False

Question 10 **10 pts**

Which lines in C++ are equivalent to this code in Java

```
int x = 5;
char a = 'A';
System.out.print( "Hello " + x + ", " + a );
```

1-
```
int x =5;
char a = 'A;
cout << "Hello %d, %c" << x << a;
```

2-
```
int x =5;
char a = 'A';
cout >> "Hello " >> x >> ", " >> a;
```
3-
```
int x =5;
char a = 'A';
cout << "Hello " << x << ", " << a;
```
4-
```
int x =5;
char a = 'A';
cout ("Hello %d, %c", x, a);
```

Answer: 3

Quiz Before Midterm *(some questions are mentioned in previous quizzes thus excluded)*

Question 3 **0.5 pts**

This programming language uses two-step translation with intermediate codes for execution.

1- C#
2- Java
3- C++
4- LISP
5- C

Answer: 2

Question 6 **0.5 pts**

Considering the following code

```
struct emp {
    int id;
    char *name;
};
struct emp john;
```

Which of the following lines are correct?

1-

```
int a = 1;
char b[ ] = "John Doe";
john.id = b;
john.name = a;
printf ("%d, %s", john.id, john.name);
```

2-

```
int a = 1;
char b[ ] = "John Doe";
john[0].id = a;
john[0].name = b;
printf ("%d, %s", john[0].id, john[0].name);
```

3- .

```
int a = 1;
char b[ ] = "John Doe";
emp.id = a;
emp.name = b;
printf ("%d, %s", emp.id, emp.name);
```

4- .

```
int a = 1;
char b[ ] = "John Doe";
john.id = a;
john.name = b;
printf ("%d, %s", john.id, john.name);
```

Answer: 4

Question 7 **0.5 pts**

Which of the following programs are correct in C? (select all that apply)

1-
```
typedef int booOoolean;
typedef char FlagType;
int main()
{
    booOoolean x = 0;
    int counter; FlagType xx = 'A'; // comment
}
```
2-
```
typedef enum { false, true } booOoolean;
typedef enum { Sun, Mon, Tue, Wed, Thu, Fri, Sat } days;
int main()
{
    booOoolean a = false;
    int counter; days x = Mon, y = Fri;
    while (x != y) x++;
}
```
3-
```
typedef enum { red, amber, green } traffic_light;
int main()
{
    traffic_light x = red;
    while (1)
        switch (x)
        {
            case amber: x = red; printf("R"); break;
            case red: x = green; printf("G"); break;
            case green: x = amber; printf("A"); break;
        }
}
```

Answer: 1,2,3

Question 8 **0.5 pts**

Which of the followings are correct declarations of the main() method (i.e., the entry point of a program)?

1-
```
int main()
```

```
    {
        return 0;
    }
2- public static void main(String[] argv)
    {
    }
3- void main()
    {
    }
4- void main(int argc, char *argv[])
    {
    }
5- int main(int argc, char *argv[])
    {
        return 0;
    }
6- main()
    {
        return 0;
    }
```

Answer: 1,3,4,5,6

Question 10 **0.5 pts**

In C, when you pass an array as a parameter to a function, what is actually passed?

1- All the values in array
2- The value of the first element in the array.
3- The address of the first element in the array
4- An array cannot be passed as a parameter to a function.

Answer: 3

Quiz Midterm

Question 1 — 7 pts

What is the output of the following codes?

1-

```
#include <stdio.h>
int main()
{
    char s1[] = "hello";
    printf("%s\n", s1);
    for (int i = 0; i < 5; i++)
        printf("%c", s1[i]);
    printf("\n");
    return 0;
}
```

Answer:

```
hello
hello
```

2-

```
#include <stdio.h>
int main()
{
    int x = 5;
    int *y;
    y = &x;
    printf("value of x: %d \n", x);
    printf("address of x: %p \n", &x);
    printf("value of y: %p \n", y);
    printf("value pointed by y: %d \n", *y);
    return 0;
}
```

Answer:

```
value of x: 5
address of x: 00000017893ff994
value of y: 00000017893ff994
value pointed by y: 5
```

3-

```
#include <stdio.h>
void fun(int x)
{
```

```
    if (x > 0) {
        printf("%d", x);
        code3(x - 1);
        printf("%d", x);
}
int main()
{
    fun(2);
    return 0;
}
```

Answer:

```
2112
```

4-

```
#include <iostream>
using namespace std;

class Queue {
public:
    Queue();

    Queue(int n);

    ~Queue();
};

Queue::Queue() {
    cout << "constructor (void)" << endl;
}

Queue::Queue(int n) {
    cout << "constructor (int)" << endl;
}

Queue::~Queue() {
    cout << "destructor" << endl;
}

int main()
{
    Queue myQueue1(500);
    Queue myQueue2;
    return 0;
}
```

Answer:

```
constructor (int)
constructor (void)
destructor
destructor
```

Question 2 **3 pts**

Write a C++ class (named Student)

```
int main()
{
    Student s(2, "John", 100); //Creating an object passing values for id,name,grade
    s.display(); // Displaying Student attributes (id, name, and grade)
    return 0;
}
```

Answer:

```
class Student
{
public:
    int id;
    string name;
    int grade;

    Student(int i, string n, int g);

    void display();
};

Student::Student(int i, string n, int g)
{
    id = i;
    name = n;
    grade = g;
}

void Student::display()
{
    cout << "id = " << id << endl;
    cout << "name = " << name << endl;
    cout << "grade = " << grade << endl;
}
```

Quiz 5

Question 1 — 10 pts

Which of the following lines (in Java, C, or C++) is equivalent to this expression in LISP?

```
(- (* 1 2) (+ 3 4) (/ 5 6) 7 )
```

1-

```
(1 * 2) - (3 + 4) - (5 / 6) – 7
```

2-

```
1 - 2 * 3 - 4 / 5 - 6 - 7
```

3-

```
(1- 2) + (3 * 4) / (5 * 6) - 7
```

4-

```
(1 - 2) * (3 - 4) * (5 - 6) / 7
```

Answer:

1

Question 2 — 10 pts

Which of the following lines represent in LISP the operation of 3 multiplied by 4?

1-

```
(multiply 3 4)
```

2-

```
(3 * 4)
```

3-

```
3 * 4
```

4-

```
(* 3 4)
```

Answer:

4

Question 3 **10 pts**

What is the value returned for the following LISP expression?

```
(+ 1   (if (< 2 1) (* 3 3)    6  ) )
```

Answer:

```
7
```

Question 4 **10 pts**

One feature of the functional paradigm is a higher level of abstraction compared with Imperative and Object-Oriented paradigms. (True/False)

Answer: True

Question 5 **10 pts**

What is the last value printed on the screen after running the following LISP code?

```
(setf num 1)
(dotimes (x 3) (setf num (+ num num) ) )
(print num)
```

Answer:

```
8
```

Question 6 **10 pts**

functions are created by calling a function-making macro. This macro is called:

1- `progn`
2- `let`
3- `defun`
4- `function`

Answer: 3

Question 7 **10 pts**

What is equivalent code in C++ for the following LISP statement:

```
(if (< 1 2) (* 3 4) (/ 5 6) )
```

1-

```
if (1 < 2) return 3 * 4; else return 5 / 6;
```

2-

```
if (1 < 2) { cout<< 3 * 4; return 5 / 6; }
```

3-

```
if (1 < 2) cout<< 3 * 4; else cout << 5 / 6;
```

4-

```
if (1 < 2) { cout<< 3 * 4; cout << 5 / 6; }
```

Answer:

1

Question 8 **10 pts**

What is printed by the print instruction in the second line?

```
(setf theList '(2 3 7 9 10) )
(print (first ( rest theList ) ) )
```

1-

```
3
```

2-

```
7
```

3-

```
2
```

4-

```
(3)
```

5-

```
(2)
```

Answer:

1

Question 9 **10 pts**

What is printed on the screen by the print instruction in the code below?

```
(print
 ( and
  (< (* 3 5 ) )
  ( not (>= 4 6 ) )
 )
)
```

Answer:

```
T
```

Question 10 **10 pts**

Is the following line correct in LISP? (True/False)

```
( + 1 2 3 4 5 6 7 8 9 0)
```

Answer: True

Quiz 6

Question 1 — 10 pts

Logic programming describes what the problem is by a set of conditions and constraints, and leaves the computer to match the problem to the existing knowledge of facts and rules and to find solutions to the problem. (True/False)

Answer: True

Question 2 — 10 pts

Which rules in Prolog match the following definition of a bad dog:

A dog is bad if it bites the postman, chews the newspaper, or chases the cat.

1-

```
is_dog(Dog), bites(Dog, Postman),is_postman(Postman):-bad_dog(Dog).
is_dog(Dog), chews(Dog, Newspaper),is_newspaper(Newspaper):-bad_dog(Dog).
is_dog(Dog), chases(Dog, Cat), is_cat(Cat):-bad_dog(Dog).
```

2-

```
bad_dog(Dog) :- is_dog(Dog); bites(Postman); is_postman(Postman).
bad_dog(Dog) :- is_dog(Dog); chews(Newspaper); is_newspaper(Newspaper).
bad_dog(Dog) :- is_dog(Dog); chases(Cat); is_cat(Cat).
```

3-

```
bad_dog(Dog) :- is_dog(Dog), bites(Dog, Postman), is_postman(Postman).
bad_dog(Dog) :- is_dog(Dog), chews(Dog, Newspaper), is_newspaper(Newspaper).
bad_dog(Dog) :- is_dog(Dog), chases(Dog, Cat), is_cat(Cat).
```

4-

```
bad_dog(Dog) :- is_dog(Dog); bites(Postman).
bad_dog(Dog) :- is_dog(Dog); chews(Newspaper).
bad_dog(Dog) :- is_dog(Dog); chases(Cat).
```

Answer: 3

Question 3 — 10 pts

```
?- 10 + 5 is 15.
```

How will prolog respond to this query? (True/False)

Answer:

False

Question 4 **10 pts**

Do these two different queries produce the same result in prolog? (True/False)

```
?- N is -(+(1,2),3).
?- N is (1+2)-3.
```

Answer: True

Question 5 **10 pts**

Given the following fact and rule.

(some extra white-spaces has been added to facilitate reading)

```
fun(1, 2).
fun(N, F) :- N>0, N1 is N - 1, fun(N1, F1), F is N * F1.
```

What is the value of the variable X after running the following query?

```
?- fun (4,X).
```

Answer:

```
48
```

Question 6 **10 pts**

Which rule defines in Prolog the following sentence:

"mary owns a Pet if it is a cat and it has black spots"

1-

```
owns(mary, Pet):- cat(Pet), black_spots(Pet).
```

2-

```
cat(Pet), black_spots(Pet):-owns(mary, Pet).
```

3-

```
mary(owns, Pet):- cat(Pet), black_spots(Pet).
```

4-

```
owns(mary, Pet):- cat(Pet); black_spots(Pet).
```

Answer:

Question 7 **10 pts**

Which of the following rules defines

" if X is instructor of the course C and Y is enrolled in the course C then X teaches Y"

1-
```
teaches(P,S) :- instructor(P,C), enrolled(S,C).
```
2-
```
instructor(P,C), enrolled(S,C) :-teaches(P,S).
```
3-
```
instructor(P,C); enrolled(S,C) :-teaches(P,S).
```
4-
```
teaches(P,S) :- instructor(P,C); enrolled(S,C).
```

Answer:

1

Question 8 **10 pts**

Which rule defines in Prolog the following sentence:

"If someone owns something, he loves it"

1-
```
loves(Who, What):-owns(Who, What).
```
2-
```
owns(Who, What):-loves(Who, What).
```
3-
```
owns(Who, What), loves(Who, What).
```
4-
```
loves(Who, What). owns(Who, What).
```
5-
```
loves(Who, What); owns(Who, What).
```

Answer:

1

Question 9 **10 pts**

Write in Prolog the question *"Which food is meal and lunch?"*

1-
```
meal(X), lunch(X).
```
2-
```
meal(X). lunch(X).
```
3-
```
meal(X):- lunch(X).
```
4-
```
meal(X); lunch(X).
```

Answer: 1

Question 10 **10 pts**

Given the following fact and rule.

(some extra white-spaces has been added to facilitate reading)

```
fun(1, 2).
fun(N, F) :- N>0, N1 is N - 1, fun(N1, F1), F is N * F1.
```

What is the value of the variable X after running the following query?

```
?- fun (1,2).
```

1- True
2- False

Answer: 1