King Saud University

College of Computer and Information Sciences
Computer Science Department

Course Code:	CSC 215
Course Title:	Procedural Programming with C
Semester:	Fall 2018
Exercises Cover Sheet:	**Mid 2 Exam**

Student Name:	
Student ID:	
Student Section No.	

Tick the Relevant	Computer Science B.Sc. Program ABET Student Outcomes	Question No. Relevant Is Hyperlinked	Covering %
√	**a) Apply knowledge of computing and mathematics appropriate to the discipline;**		
√	**b) Analyze a problem, and identify and define the computing requirements appropriate to its solution**	**6**	**40**
√	**c) Design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;**	**9**	**60**
	d) Function effectively on teams to accomplish a common goal;		
	e) Understanding of professional, ethical, legal, security, and social issues and responsibilities;		
	f) Communicate effectively with a range of audiences;		
	g) Analyze the local and global impact of computing on individuals, organizations and society;		
	h) Recognition of the need for, and an ability to engage in, continuing professional development;		
√	**i) Use current techniques, skills, and tools necessary for computing practices.**		
	j) Apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;		
	k) Apply design and development principles in the construction of software systems of varying complexity;		

Question 1: (6 marks)

a- Write the data structure needed for a doubly linked list (DLL).

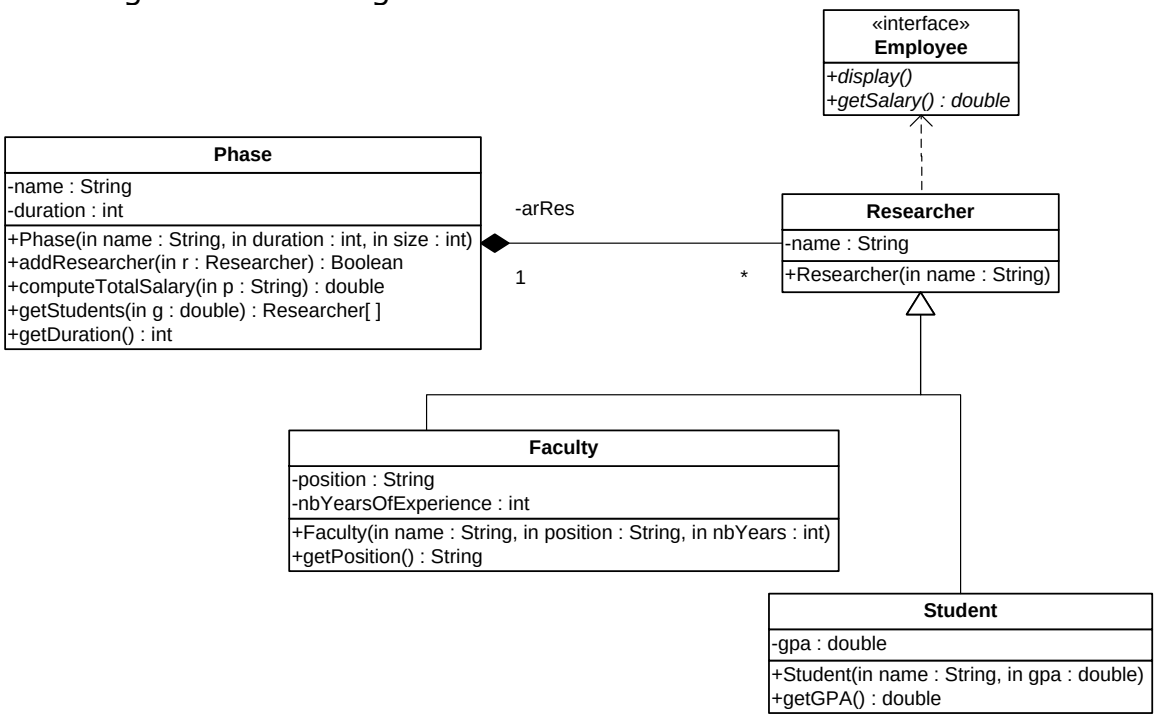
b- Write a procedure that inserts an integer in a sorted DLL.

c- Write a procedure that takes a pointer to a string and gives the number of vowels and consonants. Vowels are {a, u, i, o, e}. Consider only small characters.

Answer:

Question 2: (9 marks)

Write **clearly and neatly** the needed C header files that cope with the following UML class diagram.



Answer:

```
typedef struct Fc{
	char *position;
	int nbYearOfExperience;
}Faculty;

AssignFaculty(Faculty*, char *, int);
you do not need getter as there is no private/public field.

typedef struct std{
	double gpa;
} Student;

AssignStudent(Student*, double);

typedef struct Res{
	char *name;
	int type;
	union fs{
		Faculty f;
		Student s;
	}FS;
}Researcher;

AssignResearcherFc(Researcher*, char* name, char* pos, int nbYear, int type);  // it
calls inside AssignFaculty
```

```
AssignResearcherSt(Researcher*, char* name, double gp, int type);     // it calls inside
AssignStudent
display();

typedef struct phase{
       char *name;
       int duration;
       Researcher rc[ ];
}Phase;

AssignPhase(Phase*, char*, int, int);
int AddResearcher(Phase*, Researcher);  // the fields of researcher are already filled
double computeTotalSalary(Phase, char*);
Researcher[] getStudent(double);



//other solution

typedef struct Res{
       char *name;
}Researcher;

AssignResearcher(Researcher*, char* name);  // it calls inside AssignFaculty
displayResearcher();

typedef struct Fc{
       Researcher R ;
       char *position;
       int nbYearOfExperience;
}Faculty;

AssignFaculty(Faculty*, Researcher, char *, int);
displayFaculty();  //it calls inside displayResearcher


typedef struct std{
       Researcher R ;
       double gpa;
} Student;

AssignStudent(Student*, Researcher, double);
displayStudent();

typedef struct phase{
       char *name;
       int duration;
       Faculty fc[ ];
       Student st[ ];
}Phase;

AssignPhase(Phase*, char*, int, int);
int AddFaculty(Phase*, Faculty);  // the fields of Faculty are already filled
int AddStudent(Phase*, Student);
double computeTotalSalary(Phase, char*);
 // there is no getStudent as it is a field and we can access to it directly
```

Result					
Question No.	**Relevant Student Outcome**	**SO is Covered by %**	**Full Mark**	**Student Mark**	**Assessor's Feedback**
1	b	40	6		
2	c	60	9		
Totals		100%	15		

I certify that the work contained within this assignment is all my own work and referenced where required. **Student Signature:** **Date:**			**Feedback Received:** **Student Signature:** **Date:**	