

CPCS204, Spring 2020 Program 1

Freelance Job Marketplace

Purpose

1. Learn to implement a linked list for a real-world problem.
2. Review file I/O (input/output).

Concept Application & Algorithmic Part

Background:

Working as a freelancer has become an easy task these last few years. Previously the most difficult challenge was to find your first job or project. But now, this has become possible with a simple registration to a freelance jobs marketplace. Companies can increase their productivity by hiring skills in any field and anywhere in the world. And young freelancers can find their projects most easily. The new freelance job marketplaces offer more than just finding jobs or projects. They provide tools for accounting, managing contracts and money delivery. Some platforms even use artificial intelligence algorithms to find the best matching between offers and demands. These platforms have become a great place to build a freelancer career and a business support tool to remove borders and barriers.

The required features to be included in a freelancer platform are described as:

1. Freelancers must create their profile and describe their skills and experiences
2. Companies must post their projects and needed skills
3. Companies can search, filter and compare freelancers to select the best choice for their project
4. Enable freelancers and companies to collaborate directly with each other without third-party intervention
5. Companies can evaluate freelancers and their work

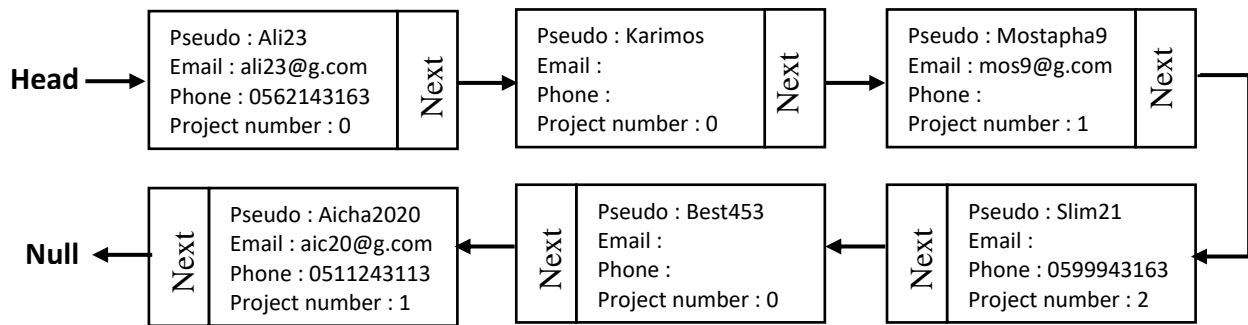


Most popular freelance job marketplaces

Question 1: (20 points)

1. Concept Application: Suppose a linked list contains the following topics of the Freelancer profiles. Write the steps to delete all incomplete profiles. A profile is incomplete if it does not contain a phone number or email address. Your answer must indicate:

- the **position of the pointer** and
- the status of the **linked list** after every step.



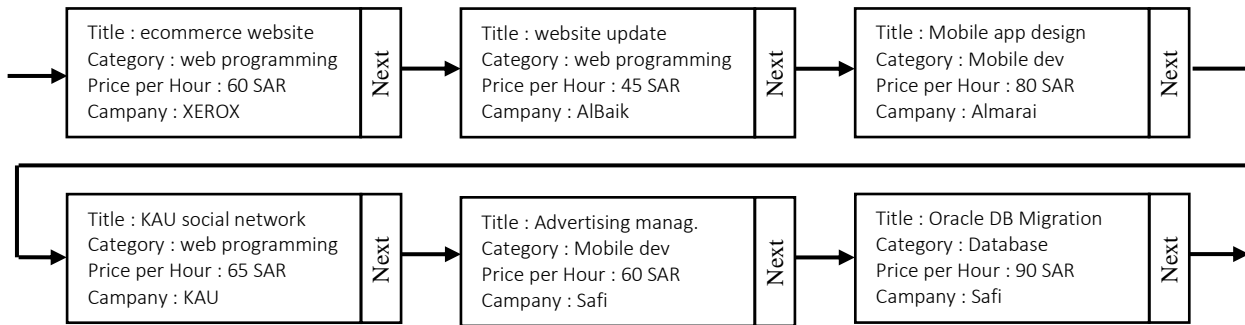
What are the steps used to delete incomplete nodes:

Question 2:(20 points)

2. Algorithm Write up: Suppose a linked list contains all available projects/jobs proposed by the companies. A freelancer wants to find all projects matching his skills and his price per hour.

Write an algorithm that searches the linked list for projects matching the freelancer's skills and the price per hour that is higher than the minimum price.

For example, a freelancer would like to search for all web programming projects with a minimum hourly price of 50 SAR. If the linked list contains the following nodes



The output of the algorithm should print

- 1/ Project: e-commerce website, web programming, 60 SAR, XEROX.
- 2/ Project: KAU social network, Web programming, 65 SAR, KAU.

Algorithm

Input:

Output:

Method:

Question 3:(60 points)

Program 2: Freelance Job Marketplace

Objective

The primary objective of this program is to implement a linked list. The secondary objective is to practice with File I/O.

Program Description

Write a program to manage a freelancer job marketplace. The first input/output file is freelancers.txt, each freelancer contains (pseudo-name, name, email, phone, skills, experience, rating, previous projects: an array of the previous projects including project Id). A freelancer can enroll in a project/job. The company must select a freelancer from among several participants in the project. Once the project is completed, the company must give feedback and a score between 0 and 5 on the freelancer's work. This score is used to calculate the freelancer's overall score.

The second file jobs.txt contains Jobs information; each job contains (project Id, company, title, description, category, min-price, max-price, status, feedback). The status of a project can be "not started", "in progress" or "completed". The minimum price and the maximum price define the price range accepted by the company.

1. Write a function to register a new freelancer and to add it to freelancers.txt.
2. Write a function to post a new project and to add it to jobs.txt.
3. Create a linked list for freelancers who have a valid account. A freelancer has a valid account if their account contains an email or phone number.
4. Write a function to find all not-started projects matching freelancer's skills and price per hour.
5. Delete a project by Project Id
6. Create a linked list for the 10 best freelancers according to their rating.

Freelancer pseudoName	name	Email	phone	skills	experience	rating	Previous project ID (Max 3)			NEXT
Anis98	Anis	Anis98@g.com	0569994523	Web design	25 projects	4.5	3	5		

Freelancer node

Project Id	company	title	description	category	min-price	max-price	status	feedback	NEXT
5	KAU	Design KAU social network	Looking for someone who can create a beautiful design for social network app with Arabic language themes	Web design	50	65	not-started		

Project node

Your program will take new freelancer information and new project information to store it in files. The program then reads a text file as input (freelancers.txt) which the information will be retrieved. The basic functionality of Freelance Job Marketplace will be implemented using two **linked lists**. The basic algorithm is as follows:

Read the freelancer text document

For each freelancer in the file

If the freelancer account is valid

Add the freelancer to the list

Read the project text document

For each project in the file, Add the project to the list

For each search attempt, check all current projects whose category matches the freelancer's skills and price.

Print the list of all projects.

Print the list of 10 best freelancers according to their rating

- To read the document use the Scanner() method
- To iterate over each freelancer or project in the documents use Scanner.next() method
- To add the freelancer to the list, create a new freelancer node, and check the availability of pseudoName, the pseudoName must be unique.
- To check the presence of a freelancer in the list, use the method checkPseudoName

```
Public boolean checkPseudoName ( String pseudo) {
    ..
    ...
}
```
- To add a project to the list, create a new project node, and check the availability of ProjectID.
- To check the presence of a project in the list, use the *check_Id* method.

```
Public boolean check_ProjectId ( String projectId) {
    ..
    ...
}
```
- To remove a project by using the projectId, the project Id must not be used in the previous project field of any freelancer.

Implementation

For this program, you will create the following classes:

- *freelancer.java*: This class will be used to create objects of type freelancer. Each freelancer object will store the pseudo-name, name, email, phone, the freelancer's experience (number of completed projects), rating and an array of the three previous completed projects.
- *project.java*: This class will be used to create objects of type project. Each project object will store the project Id, company, title, project description, category, the minimum accepted price, the maximum accepted price, status, and feedback.
- *freelanceJobMarketplace.java*: All the methods will be implemented in this class.

- *StdMenue.java*: This is the class that will contain the main. This class also prints the menu, and the user should select his option.

Input File Specifications

You will need to add a new freelancer to the file and read current information from text file. This should be automated when the user selects to add a new freelancer or select to display the current list.

Freelancers.txt:

Let's assume that the Freelancer.txt consists of the following information:

```
PseudoName ,Name,Email,Phone,Skills,Experience ,Rating,Project-ID-1,Project-ID-2,Project-ID-3#
Anis98,Anis,Anis98@kaumail.com,0536215423,web dev,5,4.5,1,3,#
Sana2000,sana,sana2000@kaumail.com, ,network administration ,1,4.7,2,,#
AbdAllah98,AbdAllah, , ,mobile dev,2,2,4,,#
```

freelancer.file.example

Projects.txt:

Let's assume that the Projects.txt consists of the following information:

```
Project Id,company,title,description,category,min-price ,max-price,status,feedback#
1,Xerox,ecommerce website,Looking for someone who can create,web dev,40,60,not started,#
2,Kau,KAU social network,We need a freelancer that ,web design ,40,65,not started,#
6,Al baik,plugin for windows desktop,We are a start up company and ,system dev,40,60,completed,Highly recommended#
```

project.file.example

The menu to be implemented are as follows

The user should select from the menu. The user will be allowed to reselect from the menu until press **8** to exit.

1. **Add a new freelancer**
2. **Add a new project**
3. **Print freelancer list with a valid account**
4. **Print 'not started' project list**
5. **Search projects for freelancer**
6. **Remove project by project Id**
7. **Print the 10 best freelancers**
8. **Exit.**

1. Add a new freelancer

This selection will allow the user to enter the freelancer information using `freelancer.java` and store the information in `freelancer.txt`. Note: the system should check whether the freelancer `pseudoName` **already exists** in the list.

Example Input:

Please enter the freelancer information:

```
PseudoName: Anis98
Name: Anis
Email: Anis98@kaumail.com
Phone: 0536215423
Skills: web dev
Experience:5
Rating: 4.5
Project-ID-1:1
Project-ID-2:2
Project-ID-3
The freelancer is added!
```

Please enter the freelancer information:

```
PseudoName: Anis98
```

```
The freelancer already exists!
```

2. Add a new project

This selection will allow the company to post a new project information using `project.java` and store the information in `projects.txt`. Note: the system should check whether the `projectId` **already exists** in the list.

Example Input:

Please enter the project information:

```
Project Id: 1
Company: Xerox
Title: e-commerce website
Description: Looking for someone who can create...
Category: web dev
min-price:40
max-price:60
```

```
The freelancer is added!
```

Please enter the project information:

Project Id: 1
This project already exists!

3. Print freelancer list with a valid account

If we use the *freelancer.file.example* file as input without adding any other freelancers, the output will be as follows

Example Output:

Pseudo	Name	Email	Phone	Skills	Exp.	Rating	Pjct-Id1	Prjct-Id2	Prjct-Id3
Anis98	Anis	Anis98@kaumail.com	0536215423	web dev	5	4.5	1	3	
Sana2000	Sana	sana2000@kaumail.com		network administration	1	4.7	2		

4. Print 'not started' project list

If we use the *project.file.example* file as input without adding any other projects, the output will be as follows

Example output:

Project Id	company	title	description	category	min-price	max-price	status	feedback
1	Xerox	Ecommerce website	Looking for someone who can create...	web dev	40	60	not started	
2	Kau	KAU social network	We need a freelancer that...	web design	40	65	not started	

Note: you have to display the **freelancer list and not started project list** in a table format, you might need `printf`.

5. Search projects for freelancer

Example Input:

Please enter the information related to the project:

skill: web design

min-price:62

Example Output:

Project Id	company	title	description	category	min-price	max-price	status	feedback
------------	---------	-------	-------------	----------	-----------	-----------	--------	----------

2	Kau	KAU social network	We need a freelancer that...	web design	40	65	not started	
---	-----	--------------------	------------------------------	------------	----	----	-------------	--

6. Remove project by project Id

Example Input:

Please enter the projectId of the project to be deleted:

Project Id: 6

The project has been deleted!

Please enter the projectId of the project to be deleted:

Project Id: 1

This project cannot be deleted!

Note: This project can't be deleted because it is listed in the freelancer Anis98's private projects.

7. Print the 10 best freelancers

Example Input:

PseudoName	Name	Email	Skills	Experience	Rating
Anis98	Anis	Anis98@kaumail.com	web dev	5	4.5
Amir	Amir	Amir@kaumail.com	web design	1	3
AbdAllah98	AbdAllah	AbdAllah98@kaumail.com	mobile dev	2	2
Ahmed	Ahmed		database	1	4
sahar	sahar	sahar@kaumail.com	linux administration	0	5
Rima96	Rima	Rima96@kaumail.com	system dev	0	4.2
sana2000	sana	sana2000@kaumail.com	network administration	0	4.7
salmankau	salman	salmankau@kaumail.com	web dev	2	3.46
Rached	Rached	Rached@kaumail.com	web dev	1	3.22
Faycel.Ali	Faycel	Faycel.Ali@kaumail.com	web dev	1	4.98
Fatma	Fatma		web design	0	3.74
Mounira	Mounira	Mounira@kaumail.com	mobile dev	0	2.5
khadija	khadija	khadija@kaumail.com	web design	3	4.26
Asma2000	Asma	Asma2000@kaumail.com	mobile dev	4	5

Example Output:

PseudoName	Name	Email	Skills	Experience	Rating
sahar	sahar	sahar@kaumail.com	linux administration	0	5

Asma2000	Asma	Asma2000@kaumail.com	mobile dev	4	5
Faycel.Ali	Faycel	Faycel.Ali@kaumail.com	web dev	1	4.98
sana2000	sana	sana2000@kaumail.com	network administration	0	4.7
Anis98	Anis	Anis98@kaumail.com	web dev	5	4.5
khadija	khadija	khadija@kaumail.com	web design	3	4.26
Rima96	Rima	Rima96@kaumail.com	system dev	0	4.2
Ahmed	Ahmed		database	1	4
Fatma	Fatma		web design	0	3.74
salmankau	salman	salmankau@kaumail.com	web dev	2	3.46

UML Diagrams:

For this program, you will create **four** Classes (UML diagram shown below):

<p><u>Freelancer</u></p> <p><i>Data Members</i></p> <pre>private String pseudoName; private String name; private String email; private String phone; private String skills; private int experience; private float rating; private int [3] prevProject; private Freelancer next;</pre> <p><i>Operations/Methods</i></p> <pre>freelancer() // one or more Constructors</pre> <p>And any other methods you need.</p>	<p><u>Project</u></p> <p><i>Data Members</i></p> <pre>private String projectId; private String company; private String title; private String description; private String category; private int minPrice; private int maxPrice; private Enumeration status; private string feedback; private Project next;</pre> <p><i>Operations/Methods</i></p> <pre>project() // one or more Constructors</pre> <p>And any other methods you need.</p>
<p><u>FreelanceJobMarketplace</u></p> <p><i>Data Members</i></p> <pre>private Freelancer headFL; private Project headPL;</pre> <p><i>Operations/Methods</i></p> <pre>checkProjectId () checkPseudoName () printValidFreelancerList () printNotStartedProject () searchProject () removeProject () printBestFreelancer ()</pre> <p>ALL necessary methods for linked-list operations</p>	<p><i>Data Members</i></p> <p>As needed</p> <p><i>Operations/Methods</i></p> <pre>public static void main()</pre> <p>ALL necessary methods for linked-list operations</p>