|  |  |
| --- | --- |
| **mkdAssignment Case** |  |
| COMP6153  Operating System |
| **Computer Science** | **O213-COMP6153-RR06-01** |
| ***Valid on*** *Odd Semester Year 2020/2021* | **Revision 00** |

1. Seluruh mahasiswa tidak diperkenankan untuk:

*All students are not allowed to:*

* + 1. Melihat sebagian atau seluruh jawaban mahasiswa lain,

*Seeing a part or the whole answer from other student*

* + 1. Menyadur sebagian maupun seluruh jawaban dari buku,

*Adapted a part or the whole answer from the book*

* + 1. Mendownload sebagian maupun seluruh jawaban dari internet,

*Downloading a part or the whole answer from the internet,*

* + 1. Mengerjakan soal yang tidak sesuai dengan tema yang ada di soal,

*Working with another theme which is not in accordance with the existing theme in the matter of the case,*

* + 1. Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + 1. Secara sengaja maupun tidak sengaja melakukan segala tindakan kelalaian yang menyebabkan hasil karyanya berhasil dicontek oleh orang lain / kelompok lain.

*Accidentally or intentionally conduct any failure action that cause the results of the project was copied by someone else / other groups.*

1. Jika mahasiswa terbukti melakukan tindakan seperti yang dijelaskan butir 1 di atas, maka **nilai mahasiswa** yang melakukan kecurangan (menyontek maupun dicontek) akan di – **NOL** – kan.

*If the student is proved to the actions described in point 1 above, the score of the student which committed dishonest acts (cheating or being cheated) will be “Zero”*

1. Perhatikan jadwal pengumpulan jawaban, segala jenis pengumpulan jawaban di luar jadwal tidak dilayani.

*Pay attention to the submission schedule, all kinds of submission outside the schedule will not be accepted*

1. Persentase penilaiaan untuk matakuliah ini adalah sebagai berikut:

*Marking percentage for this subject is described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| 40% | - | 60% |

1. Software yang digunakan pada matakuliah ini adalah sebagai berikut:

*Software will be used in this subject are described as follows:*

|  |
| --- |
| **Software**  *Software* |
| VM Ubuntu Client 20.04  Java 8  Eclipse 2020.6  NachOS 5.0j |

## Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri dan uap untuk matakuliah ini adalah sebagai berikut:

*File extensions should be included in assignment and project collection for this subject are described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| DOCX, JAVA, CLASS | - | JAVA, CLASS |

## Soal

*Case*

1. Write a command line from **home directory** to create directories with the following structure **in** **single execution**

<Home directory>

--Employee

|--Permanent Employee

| |--Teacher

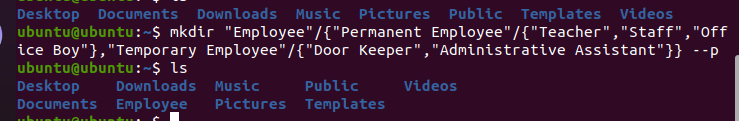
| |--Staff

| |--Office boy

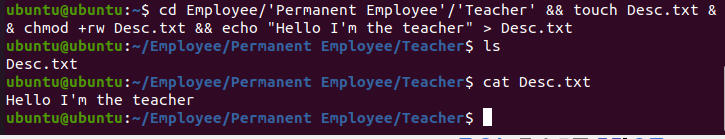
|--Temporary Employee

|--Door keeper

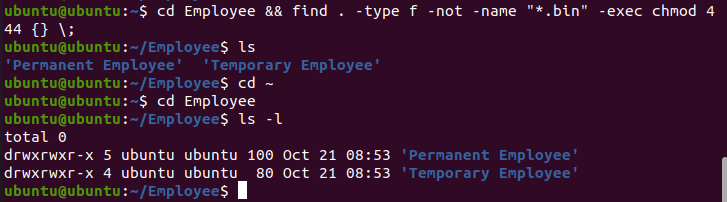
|--Administrative assistant



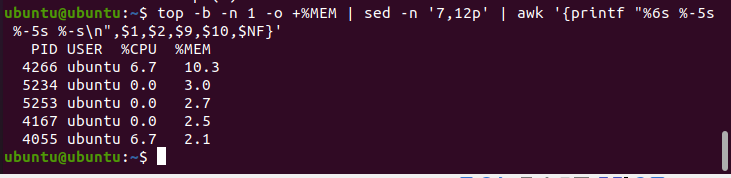
1. Write a command line to create a new file from **home directory** called ‘**Desc.txt**’ inside **Teacher** folder. Then, **write “Hello I’m the teacher”** inside the file and change the file permission to **read and write.**



1. Write a **command line** to find every **file** from **Employee** directory where the **file name** does not end with ‘**.bin**’, then change the file permission to **read** only.



1. Write a command line to show every process from **all users** and show the **process id, user, CPU, and memory usage** column, and sort the processes based on the **memory usage** in **descending order (from highest to lowest).**



**Java Programming**

**Logwarts School**

**Logwarts School** is a school that will open soon in Indonesia. But the school still need more employees and you are willingly to create a program to handle new employee data for the school.

Here are the detail requirements of the application:

* The application must use **Object-Oriented Programming** concepts such as **Encapsulation**, **Inheritance**, and **Polymorphism**.There must be **Polymorphic Classes** and at least **1 Overridden Method**.
* Create a method to count the employee **total** **salary** with the following formula:

|  |  |  |
| --- | --- | --- |
| **Position** | **Total Salary (Employee)** | **Total salary (Permanent Employee)** |
| Teacher | Total salary = salary + (salary \* 20%) | Total salary = salary + (salary \* 20%) + bonus salary |
| Staff | Total salary = salary + (salary \* 15%) | Total salary = salary + (salary \* 15%) + bonus salary |
| Office boy | Total salary = salary | Total salary = salary + bonus salary |

* When the **program starts**, the user will be given three choices:
  + **Insert Employee**
  + **Insert Permanent Employee**
  + **Exit**

A close up of a logo

Description automatically generated

**Figure 1. Screenshot of the Main Menu**

* If the user selects the firstmenu (Insert Employee), then:
* Ask the user to input **employee name**. The name must start with ‘**Mr.**’ or ‘**Mrs.**’
* Ask the user to input **employee address**, the employee address must end with ‘**Street**’
* Ask the user to input **employee gender**, the employee gender must between ‘**Male**’ or ‘**Female**’
* Ask the user to input **employee position**, the employee position must either ‘**Teacher**’, ‘**Staff**’, or ‘**Office boy**’
* Ask the user to input **employee salary**, the **minimum** employee salary should be **4000000** and **have no maximum salary**

A close up of text on a white background

Description automatically generated

**Figure 2. Screenshot of input Employee Information**

* + Show all the inputted information from the user

A screenshot of a cell phone

Description automatically generated

**Figure 3. Screenshot of displaying Employee Information after input**

* If the user selects the secondmenu (Insert Permanent Employee), then:
* Ask the user to input **employee name**. The name must start with ‘**Mr.**’ or ‘**Mrs.**’
* Ask the user to input **employee address**, the employee address must end with ‘**Street**’
* Ask the user to input **employee gender**, the employee gender must between ‘**Male**’ or ‘**Female**’
  + Ask the user to input **employee position**, the employee position must either ‘**Teacher**’, ‘**Staff**’, or ‘**Office boy**’
  + Ask the user to input **employee salary**, the **minimum** employee salary should be **4000000** and **have no maximum salary**
  + Ask the user to input **bonus salary**, the bonus salary must be **greater than** **50000**

A close up of a newspaper

Description automatically generated

**Figure 4. Screenshot of displaying Employee Information after input**

* + Show all inputted information from the user

A screenshot of a cell phone

Description automatically generated

**Figure 5. Screenshot of displaying Employee Information after input**

* If the user selects the third menu (Exit), **terminate** the program immediately
* After the user finish entering the employee data, return the user to **Main Menu**