

Case Study AY 2021-2022
File Handling

Create a file for the defined set of employee records that will have the following fields:

Employee Structure

Employee Number	10 characters
Employee Name	15 characters
Status Code	character (R or C)
Hours Worked	999
Deductions	99999.99

The program will have the following requirements:

1. Created file may be in text or binary file processing modes.
2. Input the required data for the following fields for n employees.
3. Status code R is for Regular and C is for Casual.
4. If status code is R, input the basic salary amount and check if the employee exceeded 160 hours of work for the month. If employee exceeded 160 hours of work for the month, basic rate is basic pay amount divided by 160 hours. Overtime rate is half more than his/her basic rate. Compute for Overtime pay.
Note: Overtime hour/s is hours worked > 160.
5. If status code is C, then input basic rate. Basic salary amount is equivalent to number of hours worked multiplied by basic rate. If hours worked exceeds 160 hours then the employee is entitled to an Overtime pay which is half more than his/her basic rate. Compute for Overtime pay.
Note: Overtime hour/s is hours worked > 160.
6. Compute for Net pay using the formula
Net pay = Basic Pay + Overtime pay – Deductions
7. Produce the output layout below:

Sample output layout:

ABC COMPANY Makati City						
Payroll						
Employee Number	Employee Name	Status	Basic Salary	Overtime Pay	Deductions	Net Pay
1234567890	Juan dela Cruz	Regular	30000.00	1500.00	2356.00	29144.00
2345678901	Maria Makiling	Casual	15000.00		1500.00	13500.00
.						
.						

Note: If Status code is equal to R then display the word “Regular”.
If Status code is equal to C then display the word “Casual”.

Case Study AY 2021-2022

Linked List

Create a menu driven program which will have the following choices:

Example:

MAIN MENU

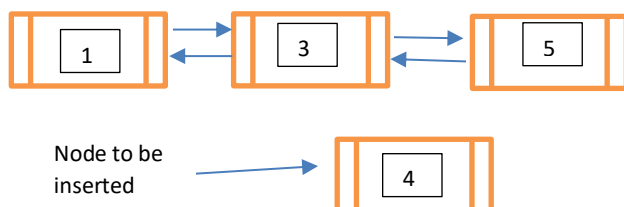
1. Create a Doubly Linked List
2. Display a Doubly Linked List
 21. Normal order
 22. Reverse order
3. Sort a Doubly Linked List
 31. Ascending
 32. Descending
4. Insert a Node
5. Delete a Node
6. Search a Number in the List
7. Exit

Enter choice: ____

Requirements:

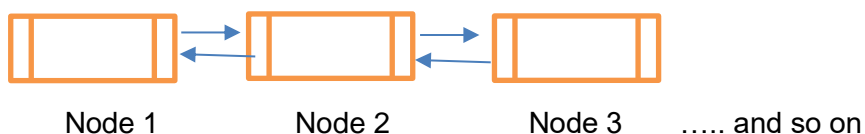
- Node structure will contain the following fields:
Pointer prev
Integer x
Pointer next
- The Main Menu should be displayed after the execution of a choice unless the Exit option is chosen.
- Create a Doubly Linked List first to enable the other options. You cannot choose the other options (except the Exit option) unless you have created the linked list.
- In the Insert a Node option, you have to input the number you want to insert and insert it using the sorted list.

Example:



The number 4 should be inserted in between 3 and 5. You CANNOT add the number 4 at the end and then sort the list.

- In the Delete a node option, input a number to be deleted and delete that node. When the number does not exist, then display a message stating that the number does not exist.
- In the Search a number option, user is prompted to enter a number to be searched until user enters 'N' or 'n' in the prompt "Do you want to search another number? "
If the number searched is in the list, then display the message " <number> is in the list. It is in node number/s _____. If there are multiple occurrences of the number to be searched then all node numbers should be listed. Otherwise, display the message " <number> is not in the list".



- You may add other features which you think is necessary

HAPPY PROGRAMMING!!! GOD BLESS!!!