Essay

$$\frac{n}{2}(2a + (n-1)b)$$

1. (10 points) Convert to prefix notation using a stack:
2. (10 points) Compute the prefix evaluation result using a stack for n = 6, a = 3, b = 2
3. (5 points) Create an expression tree and write its postorder traversal result

Case study hash table

1. (20 points) Complete the following code snippet.

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
#define SIZE 50

struct Data{
        char coin[5];
        double price;
        Data *next;
} *arr[SIZE];

int midSquare(char coin[]){
        //Mid square function
}

void insert(char coin[], double price){
        Data *data = (Data*)malloc(sizeof(Data));
        data->price = price;
        strcpy(data->coin, coin);
        data->next = NULL;

        int index = midSquare(coin);
        //Insert with chaining collision handling
}

int main(){
        insert("BTC", 97000.00);
        insert("ETH", 2700.00);
        insert("SOL", 172.59);
        insert("XRP", 2.57);

        return 0;
}
```

2. (5 points) Complete the following table based on the code you have written.

Coin	Price	Index
BTC	$97,000	
ETH	$2,700	
SOL	$172.59	
XRP	$2.57	

Case Study Priority Queue

Copi Xpress is a company operating in the logistics sector. To simplify package delivery data management for couriers, Copi Xpress intends to develop a shipment recording system using the priority queue concept. As a programmer, your task is to create a program for Copi Xpress.

```
Copi Xpress
1. Insert new package
2. Send a package
3. View all packages
4. Exit
>>
```

Figure 1. Main Menu

1. Insert new package
    - The user will be required to input:
        - Address (must start with "Jl.", 5–20 characters [inclusive])
        - Random ID with the format:

CPX[A-Z,0-9][A-Z,0-9][A-Z,0-9]
Example: CPX0J2

        - Package Type (Economy/Standard/Instant [case sensitive])
    - Insert the data into a **linked list** and show success message.

```
>> 1
Address (must start with "Jl.", 5-20 characters [inclusive]): Sentul
Address (must start with "Jl.", 5-20 characters [inclusive]): Jl. Sentul
Package Type (Economy/Standard/Instant [case sensitive]): instant
Package Type (Economy/Standard/Instant [case sensitive]): Instant
Package inserted
```

Figure 2. Insert new package

2. Send a package
    - The package shipping order follows the priority: **Instant → Standard → Economy**. Display the following data:
    - Remove the package data from the **linked list** once it has been sent.
    - If no packages are available, show **"No data"** and return to the main menu

```
>> 2
No data
```

Figure 3. No Package Available to send

```
>> 2
CPXEAI (Instant) successfully sent to Jl. Sentul
```

Figure 4. Sending a Package

3. View all packages
    - Display all available package data.
    - If no packages are available, show **"No data"** and return to the main menu

```
>> 3
No data
```

Figure 5. No Package Available

```
>> 3
CPXEAI - Instant - Jl. Sentul
CPXFED - Instant - Jl. Foresta
CPX7BI - Standard - Jl. Anggrek
CPX330 - Economy - Jl. Kb Jeruk
CPXDEJ - Economy - Jl. WSA
```

Figure 6. View All Available Packages to Send

4. Exit
    - Send all remaining packages, then exit the program.

```
>> 4
CPXFED (Instant) successfully sent to Jl. Foresta
CPX7BI (Standard) successfully sent to Jl. Anggrek
CPX330 (Economy) successfully sent to Jl. Kb Jeruk
CPXDEJ (Economy) successfully sent to Jl. WSA
Exit Program
```

Figure 7. Send All Packages & Exit

Scoring (50 points):

1. (2 points) Program running well & proper menu navigation
2. (10 points) Successfully insert new packages based on priority concept
3. (8 points) Successfully removed a package from linked list
4. (2 points) Successfully exit program and remove all available packages
5. (8 points) Successfully display all packages
6. Validation:
    - (4 points) Address
    - (4 points) Package Type
    - (4 points) Valid success message for insert and send
    - (4 points) Display no data & return to the main menu if no data is available
7. (4 points) Generate random Id