Stock Portfolio Tracker

```
class StockPortfolio:
    def __init__(self):
        # Initialize an empty dictionary to hold stock data
        self.portfolio = {}

    def add_stock(self, stock, shares, purchase_price):
        # Add stock to the portfolio
        if stock in self.portfolio:
            print(f"{stock} already exists in the portfolio. Use update to change shares or price.")
        else:
            self.portfolio[stock] = {'shares': shares, 'purchase_price': purchase_price}
            print(f"Added {shares} shares of {stock} at ${purchase_price} each.")

    def remove_stock(self, stock):
        # Remove stock from the portfolio
        if stock in self.portfolio:
            del self.portfolio[stock]
            print(f"Removed {stock} from the portfolio.")
        else:
            print(f"{stock} not found in the portfolio.")

    def update_current_price(self, stock, current_price):
        # Update the current price of a stock
        if stock in self.portfolio:
            self.portfolio[stock]['current_price'] = current_price
            print(f"Updated {stock} current price to ${current_price}.")
        else:
            print(f"{stock} not found in the portfolio.")
```

```
    def performance(self):
        # Calculate performance of the portfolio
        print("\nPortfolio Performance:")
        for stock, data in self.portfolio.items():
            shares = data['shares']
            purchase_price = data['purchase_price']
            current_price = data.get('current_price', 0)
            profit_loss = (current_price - purchase_price) * shares
            print(f"{stock}: {shares} shares, Purchase Price: ${purchase_price}, Current Price: ${current_price}, Profit/Loss: ${profit_loss}")

def main():
    portfolio = StockPortfolio()

    while True:
        print("\nOptions:")
        print("1. Add Stock")
        print("2. Remove Stock")
        print("3. Update Current Price")
        print("4. Show Portfolio Performance")
        print("5. Exit")

        choice = input("Choose an option (1-5): ")

        if choice == '1':
            stock = input("Enter stock symbol: ")
            shares = int(input("Enter number of shares: "))
            purchase_price = float(input("Enter purchase price per share: "))
            portfolio.add_stock(stock, shares, purchase_price)

        elif choice == '2':
```

```
            stock = input("Enter stock symbol to remove: ")
            portfolio.remove_stock(stock)

        elif choice == '3':
            stock = input("Enter stock symbol to update: ")
            current_price = float(input("Enter current price: "))
            portfolio.update_current_price(stock, current_price)

        elif choice == '4':
            portfolio.performance()

        elif choice == '5':
            print("Exiting the program.")
            break

        else:
            print("Invalid option. Please choose again.")

if __name__ == "__main__":
    main()
```

OUTPUT:

```
Options:
1. Add Stock
2. Remove Stock
3. Update Current Price
4. Show Portfolio Performance
5. Exit
Choose an option (1-5): 1
```

Enter stock symbol: @

Enter number of shares: 100

Enter purchase price per share: 50

Added 100 shares of @ at $50.0 each.

Options:

1. Add Stock

2. Remove Stock

3. Update Current Price

4. Show Portfolio Performance

5. Exit

Choose an option (1-5): 1

Enter stock symbol: #

Enter number of shares: 50

Enter purchase price per share: 100

Added 50 shares of # at $100.0 each.

Options:

1. Add Stock

2. Remove Stock

3. Update Current Price

4. Show Portfolio Performance

5. Exit

Choose an option (1-5): 3

Enter stock symbol to update: @

Enter current price: 100

Updated @ current price to $100.0.

Options:

1. Add Stock

2. Remove Stock

3. Update Current Price

4. Show Portfolio Performance

5. Exit

Choose an option (1-5): 4

Portfolio Performance:

@: 100 shares, Purchase Price: $50.0, Current Price: $100.0, Profit/Loss: $5000.0

#: 50 shares, Purchase Price: $100.0, Current Price: $0, Profit/Loss: $-5000.0

Options:

1. Add Stock

2. Remove Stock

3. Update Current Price

4. Show Portfolio Performance

5. Exit

Choose an option (1-5): 5

Exiting the program.

=== Code Execution Successful ===