```
# Define an empty list to store tasks
tasks = []

# Function to display the to-do list
def display_tasks():
    if not tasks:
        print("Your to-do list is empty.")
    else:
        print("To-Do List:")
        for i, task in enumerate(tasks, start=1):
            status = "Done" if task["completed"] else "Not Done"
            print(f"{i}. {task['task']} ({status})")

# Function to add a task to the to-do list
def add_task(task_name):
    task = {"task": task_name, "completed": False}
    tasks.append(task)
    print(f"Task '{task_name}' added to your to-do list.")

# Function to mark a task as completed
def mark_completed(task_number):
    if 1 <= task_number <= len(tasks):
        tasks[task_number - 1]["completed"] = True
        print(f"Task {task_number} marked as completed.")
    else:
        print("Invalid task number. Please enter a valid task number.")

# Function to remove a task from the to-do list
def remove_task(task_number):
    if 1 <= task_number <= len(tasks):
        removed_task = tasks.pop(task_number - 1)
```

```
        print(f"Task '{removed_task['task']}' removed from your to-do list.")
    else:
        print("Invalid task number. Please enter a valid task number.")

# Main program loop
while True:
    print("\nOptions:")
    print("1. Display to-do list")
    print("2. Add a task")
    print("3. Mark a task as completed")
    print("4. Remove a task")
    print("5. Quit")
    choice = input("Enter your choice: ")

    if choice == '1':
        display_tasks()
    elif choice == '2':
        task_name = input("Enter the task: ")
        add_task(task_name)
    elif choice == '3':
        display_tasks()
        task_number = int(input("Enter the task number to mark as completed: "))
        mark_completed(task_number)
    elif choice == '4':
        display_tasks()
        task_number = int(input("Enter the task number to remove: "))
        remove_task(task_number)
    elif choice == '5':
        break
    else:
        print("Invalid choice. Please enter a valid option.")
```