EXPERIMENT NO.03

Name:Shivani kolekar
Roll no.24141031
Batch:I2

Title: knapsack problem

Program example

```
#include <stdio.h>
int max(int a, int b) {
    return (a > b) ? a : b;
}

// Recursive function to solve knapsack
int knapsack(int W, int wt[], int val[], int n) {
    // Base case: no items or no capacity
    if (n == 0 || W == 0)
        return 0;

    // If weight of the nth item is more than capacity, skip it
    if (wt[n - 1] > W)
        return knapsack(W, wt, val, n - 1);

    // Return max of two cases:
    // (1) nth item included
    // (2) not included
    else
        return max(
            val[n - 1] + knapsack(W - wt[n - 1], wt, val, n - 1),
            knapsack(W, wt, val, n - 1)
        );
}

int main() {
    int val[] = {60, 100, 120};
    int wt[] = {10, 20, 30};
    int W = 50;
    int n = sizeof(val) / sizeof(val[0]);

    int result = knapsack(W, wt, val, n);
    printf("Maximum value in knapsack = %d\n", result);

    return 0;
}
```