LAB ASSIGNMENT – 04

Program

```
#include <stdio.h>
#include <stdlib.h>

int max(int a, int b) {
    return (a > b) ? a : b;
}

int knapsack(int W, int wt[], int val[], int n) {
    int i, w;
    int **dp = (int**)malloc((n+1) * sizeof(int*));
    // Allocate memory for DP table
    for (i = 0; i <= n; i++)
        dp[i] = (int*)malloc((W+1) * sizeof(int));
    // Build table dp[][] in bottom-up manner
    for (i = 0; i <= n; i++) {
        for (w = 0; w <= W; w++) {
            if (i == 0 || w == 0)
                dp[i][w] = 0;
            else if (wt[i-1] <= w)
                dp[i][w] = max(val[i-1] + dp[i-1][w-wt[i-1]], dp[i-1][w]);
            else
                dp[i][w] = dp[i-1][w];
        }
    }
    int result = dp[n][W];
    int max_val = result; // Store result before backtracking

    // Print which items are selected
    printf("Selected items: ");
    w = W;
    for (i = n; i > 0 && result > 0; i--) {
        if (result != dp[i-1][w]) {
            printf("%d ", i);
```

```
            result -= val[i-1];
            w -= wt[i-1];
        }
    }
    printf("\n");
    // Free allocated memory
    for (i = 0; i <= n; i++)
        free(dp[i]);
    free(dp);
    return max_val;
}

int main() {
    int n, W;
    printf("Enter number of items: ");
    scanf("%d", &n);
    printf("Enter knapsack capacity: ");
    scanf("%d", &W);
    int *val = (int*)malloc(n * sizeof(int));
    int *wt = (int*)malloc(n * sizeof(int));

    printf("Enter values and weights:\n");
    for (int i = 0; i < n; i++) {
        printf("Item %d - Value: ", i+1);
        scanf("%d", &val[i]);
        printf("Item %d - Weight: ", i+1);
        scanf("%d", &wt[i]);
    }
    int max_value = knapsack(W, wt, val, n);
    printf("Maximum value: %d\n", max_value);

    free(val);
    free(wt);
    return 0;
}
```

OUTPUT

```
(base) PS C:\Users\Karunya\Documents\Sem 7 - LAs\DAA\execution_daa> .\knapsack_01.exe
Enter number of items: 3
Enter knapsack capacity: 50
Enter values and weights:
Item 1 - Value: 60
Item 1 - Weight: 10
Item 2 - Value: 100
Item 2 - Weight: 20
Item 3 - Value: 120
Item 3 - Weight: 30
Selected items: 3 2
Maximum value: 220
(base) PS C:\Users\Karunya\Documents\Sem 7 - LAs\DAA\execution_daa>
```