01.11.2024

Istanbul Technical University
Department of Computer Engineering

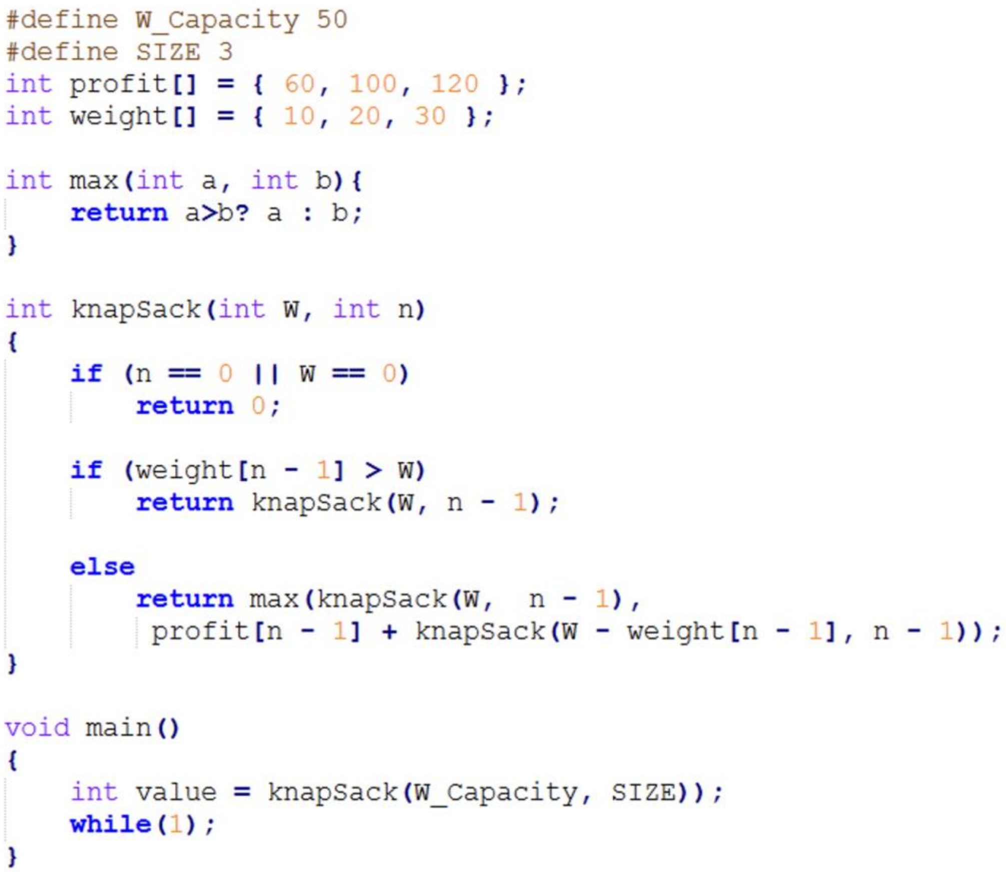
BLG 212E – Microprocessor Systems
2024-2025 Homework 1

Due Date: 10.11.2024, **Sunday,** 23:59.

Question 1 (100 Points):

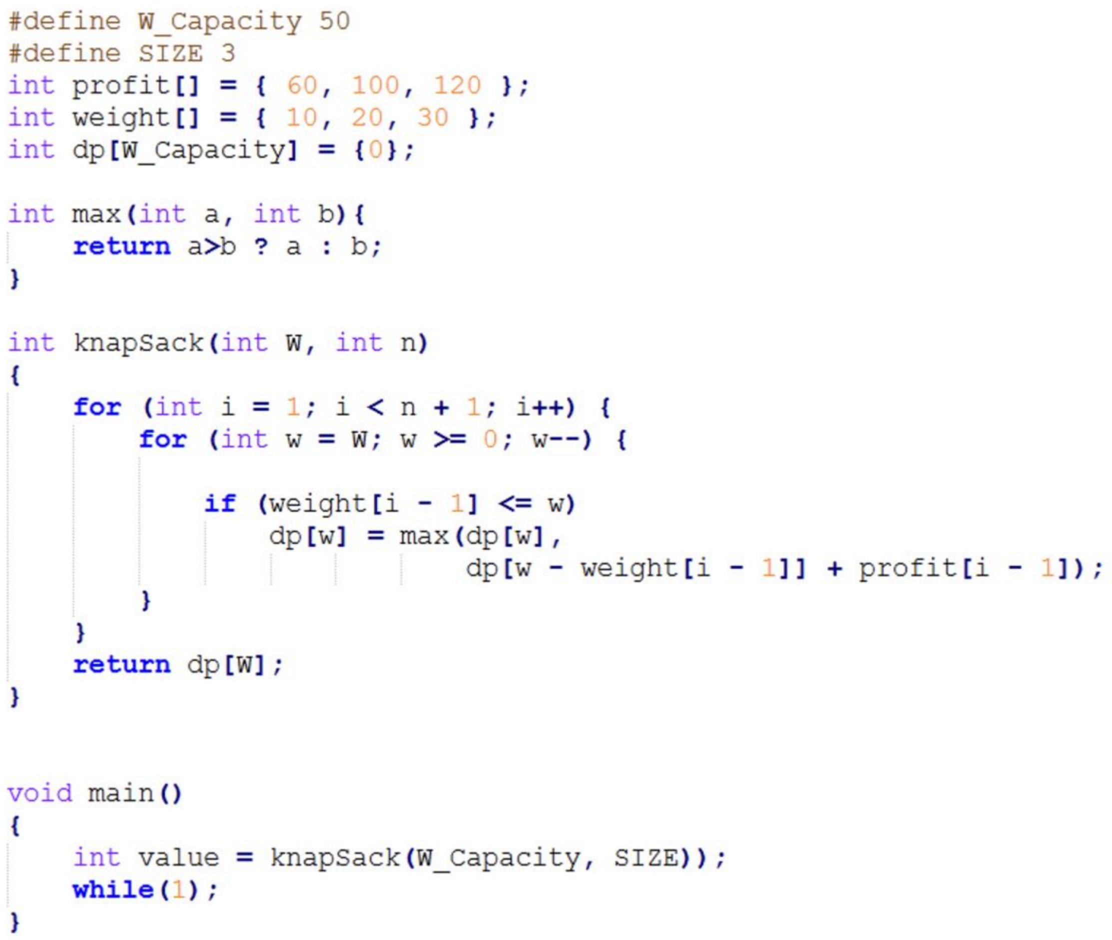
You are asked to implement the 0/1 Knapsack Problem using the ARM Cortex M0+ architecture. In this problem, each item has a weight (W) and a profit (P). Our knapsack has a limited weight capacity (W). The objective is to maximize the profit without exceeding the weight capacity of the knapsack.

a) (50 Points) You are asked to implement the 0/1 knapsack problem recursively as given below.



```
#define W_Capacity 50
#define SIZE 3
int profit[] = { 60, 100, 120 };
int weight[] = { 10, 20, 30 };

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

int knapSack(int W, int n)
{
    if (n == 0 || W == 0)
        return 0;

    if (weight[n - 1] > W)
        return knapSack(W, n - 1);

    else
        return max(knapSack(W,  n - 1),
         profit[n - 1] + knapSack(W - weight[n - 1], n - 1));
}

void main()
{
    int value = knapSack(W_Capacity, SIZE));
    while(1);
}
```

b) (50 Points) You are asked to implement the 0/1 knapsack problem iteratively as given below.



```
#define W_Capacity 50
#define SIZE 3
int profit[] = { 60, 100, 120 };
int weight[] = { 10, 20, 30 };
int dp[W_Capacity] = {0};

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

int knapSack(int W, int n)
{
    for (int i = 1; i < n + 1; i++) {
        for (int w = W; w >= 0; w--) {

            if (weight[i - 1] <= w)
                dp[w] = max(dp[w],
                            dp[w - weight[i - 1]] + profit[i - 1]);
        }
    }
    return dp[W];
}


void main()
{
    int value = knapSack(W_Capacity, SIZE));
    while(1);
}
```

Constraints:

- Your main function name or label must be "__main".
- Your code should include a comment for each line. Otherwise, points will be deducted.
- The program must be implemented with Arm Cortex M0+ assembly language.
- Your assembly source file is expected to work with Keil µVision IDE v5.
- Default configuration must be sufficient to run your programs. If your program expects any different configuration parameter, please write this at the top of the code in comment lines.
- If your program does not run with Keil µVision IDE you will get zero point from this question.
- At the end of the program, ensure that R0 stores the value variable from the main function, R1 stores the starting address of the profit array, R2 stores the starting address of the weight array, and R3 stores the address of the dp array (if available).

Submission: Please submit an assembly file for Question Q1.a and another assembly file for Question 1.b. Type your name and student ID at the top of files as comments. You are expected to submit your homework through the Ninova system before the due date. Late submissions will not be accepted.

Any solution must be your own work. If any plagiarism is detected, disciplinary regulations of the university will be followed.

Note: If you have any questions regarding the exam, you may contact to teaching assistant of the course. (Kadir Özlem (kadir.ozlem@itu.edu.tr)).