جامعة بيرزيت
BIRZEIT UNIVERSITY

ENCS4370- Computer Architecture
Summer Semester 2022/2023

Project#1

Implementing Knapsack Problem in MIPS Assembly

Deadline: August 23, 2023

Problem Definition

Let $\mathbf{U = \{u_1, u_2, u_3, \ldots, u_n\}}$ a set of items to be packed in a knapsack of size **C**. for $\mathbf{1 \leq i \leq n}$, let $\mathbf{s_i}$ and $\mathbf{v_i}$ are the size and the value of the $\mathbf{i^{th}}$ item, respectively. The objective is to fill the knapsack with some items from **U**, such that the total size of these items does not exceed the knapsack size **C** and the total value of these items is maximum. Assume the size of each item is less than **C**.

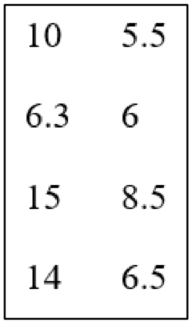
Hint: Sort the items in a descending order based on their value to size ratio, i.e. $\mathbf{v_i / s_i}$. Then consider each item one by one for packing. If the size of the current item fits in the available space of the knapsack, pack it. Otherwise, skip it and go to the next item. The program stops either when all items have been considered or no more items can be packed.

In this project, you are required to implement this Knapsack problem in MIPS assembly language using MARS simulator.

Input:

1. **Knapsack size:** the program prompts the user to enter the knapsack size which is an integer value
2. **List of items in a text file:** each item is represented by two floating point numbers. The first one represents the item size and the second one represents the item value. Thus, the input file should contain an even number of these numbers. Moreover, no item size should be greater than the knapsack size.
3. The program proceeds if all input data are valid. Otherwise, the program prints an error message and terminates

The input file will look like this

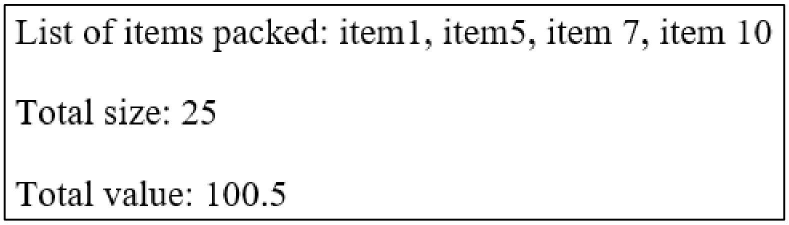


```
10     5.5

6.3    6

15     8.5

14     6.5
```

Output:

1. Subset of items that have been packed
2. The total size of these items
3. The total value of these items

The program should print its output in a text file as follows:



```
List of items packed: item1, item5, item 7, item 10

Total size: 25

Total value: 100.5
```

4. **Program Menu (Program Flow):**

The following menu is displayed to the user in an infinite loop.

1. The program asks the user to enter the knapsack size
2. If the entered number is valid, the program proceeds. Otherwise, it prints an error message and terminates
3. The program asks the user to enter the path of the input file
4. If the file exists and contains valid data, the program proceeds and packs the items into the knapsack. Otherwise, it prints an error message and terminates
5. Finally, when the program completes, it prints a message to the user and prints the results on the output file

Teamwork:

You can work on this project in teams of up to two students only

Submission

You need to submit the complete MIPS assembly files

Grading Criteria

Criteria	**Grade**
Code Structure, Organization, and Documentation	10
Discussion	10
User Interface (Menu)	10
Program Running Properly	20
Reading/Writing from/to Text Files	10
Computing size/value ratios	10
Descending order of items	20
Filling the knapsack properly	20
Input data validation	10
Total	**120**