

Programming assignment #2

Farming Problem

Objective

1. To understand how to design an algorithm.
2. To exercise the concept of recursion and dynamic programming.
3. To learn how to analyze an algorithm.

Problem

You are a farmer who has a field with area A. There are N kinds of crops and their corresponding areas and values. Please design algorithms with recursive, dynamic programming, and memorized recursive strategy, respectively, to evaluate the highest benefit with following rules:

1. Each crop can be planted more than once.
2. The area of the crop is indivisible. (e.g. one crop with 3 units area and 6 units value, you cannot plant 2 units area for 4 units value. 3 units is the least area for this crop.)

Compile & Execute

Compile command : `$ g++ <hw2.cpp> -O3`

Execute command : `$./<execute file> <input file> <output file>`

(Ex : `$./a.out input1.txt output1.txt`)

Note that input and output file should be the arguments of program. Please make sure your code can be compiled and executed.

Input

Input file is a node list of following format :

```
Mode
<Field Area> <# of crop kinds>

// crop list
<crop area> <crop value>
```

The first line is the algorithm mode, 0 for recursive, 1 for dynamic programming, and 2 for memorized recursive. The second line includes your field area and the total number of crop kinds. In the crop list, each line represents a kind of crop and includes area and value.

Input example

```
1
50 7

31 69
10 20
21 39
19 37
4 7
3 5
6 11
```

Output

```
106      // highest benefit
```

Program Submission

1. Please use the C++ language and your program **must be written in only one source file**.
2. Your source file must be named as “**Student_ID_number_hw2.cpp**” and please make sure that all characters of the filename are in lower case. For example, if your student number is 9711592, the name of your program file should be “**9711592_hw2.cpp**”.
3. Upload your report and program to the E3 by deadline.
4. Don't print any words on the terminal.

Report

1. No more than 4 pages. (including bonus)
2. Your report must contain:
 - a. The flow chart or the pseudo code of your program. (Do **NOT** use handwriting picture or screenshot of entire program.)
 - b. The design concept of your algorithm.
 - c. The comparison among three kinds of implementation.
 - d. Problems in implement time and discussion. (optional)
 - e. bonus part (optional)
3. The report file name must be “**Student_ID_number_hw2.doc(x)**” or “**Student_ID_number_hw2.pdf**” and please make sure that all characters of the filename are in lower case. For example, if your student number is 9711592, the name of your program file should be “**9711592_hw2.pdf**”.

Bonus

There are several auxiliary cases. Please analyze the time and space complexity of these auxiliary cases, and compare the results with original case. (5%) Moreover, please design a dynamic programming algorithm to solve the auxiliary cases in the same time complexity of original case and explain your method. (10%) Please write the bonus part in your report with an independent paragraph.

Grading

- | | |
|-----------------------|------------|
| • Report | 25% |
| • Case 1 | 15% |
| • Case 2 | 15% |
| • Hidden cases | 30% |
| • Bonus | 15% |

* Time limit for dynamic programming and memorized recursive version is 300s.

Noting

- **Due Date : 2019/12/13 23:55:00**
- **You'll get 0 points if not hand in on time.**
- **Plagiarism is forbidden.**