نام و نام خانوادگی: شماره دانشجویی(لاتین): زمان: 11110 دقیقه

هر سوال را در محل در نظر گرفته شده پاسخ دهید. پاسخ های خارج از محل تصحیح نمیشوند. شماره دانشجویی باید با **اعداد لاتین** نوشته شود.

سارا کدیری ـ محمد جواد میرشکاری حقیقی ـ سید صالح اعتمادی

1. [50] You work for a small manufacturing company and have recently been placed in charge of shipping items from the factory, where they are produced, to the warehouse, where they are stored. Every day the factory produces n items which we number from 1 to n in the order that they arrive at the loading dock to be shipped out. As the items arrive at the loading dock over the course of the day they must be packaged up into boxes and shipped out. Items are boxed up in contiguous groups according to their arrival order; for example, items $1 \dots 6$ might be placed in the first box, items $7 \dots 10$ in the second, and $11 \dots 42$ in the third. Items have two attributes, *value* and *weight*, and you know in advance the values $v_1 \dots v_n$ and weights $w_1 \dots w_n$ of the items. There are two types of shipping options available to you:

   Limited-Value Boxes: One of your shipping companies offers insurance on boxes and hence requires that any box shipped through them must contain no more than V units of value. Therefore, if you pack items into such a "limited-value" box, you can place as much weight in the box as you like, as long as the total value in the box is at most V.

   Limited-Weight Boxes: Another of your shipping companies lacks the machinery to lift heavy boxes, and hence requires that any box shipped through them must contain no more than W units of weight. Therefore, if you pack items into such a "limited-weight" box, you can place as much value in the box as you like, as long as the total weight inside the box is at most W.

   Please assume that every individual item has a value at most V and a weight at most W. You may choose different shipping options for different boxes. Your job is to determine the optimal way to partition the sequence of items into boxes with specified shipping options, so that shipping costs are minimized.

   Suppose limited-value and limited-weight boxes each cost \$1 to ship. Describe an $O(n)$ greedy algorithm that can determine a minimum-cost set of boxes to use for shipping the items. Justify why your algorithm produces an optimal solution.

   We use a greedy algorithm that always attempts to pack the largest possible prefix of the remaining items that still fits into some box, either limited-value or limited-weight. The algorithm scans over the items in sequence, maintaining a running count of the total value and total weight of the items encountered thus far. As long as the running value count is at most V or the running weight count is at most W, the items encountered thus far can be successfully packed into some type of box. Otherwise, if we reach a item j whose value and weight would cause our counts to exceed and , then prior to processing item we first package up the items scanned thus far (up to item $j-1$) into an appropriate box and zero out both counters. Since the algorithm spends only a constant amount of work on each item, its running time is $O(n)$. Why does the greedy algorithm generate an optimal solution (minimizing the total number of boxes)? Suppose that it did not, and that there exists an optimal solution different from the greedy solution that uses fewer boxes. Consider, among all optimal solutions, one which agrees with the greedy solution in a maximal prefix of its boxes. Let us now examine the sequence of boxes produced by both solutions, and consider the first box where the greedy and optimal solutions differ. The greedy box includes items $i \dots j$ and the optimal box includes items $i \dots k$, where $k < j$ (since the greedy algorithm always places the maximum possible number of items into a box). In the optimal solution, let us now remove items $k+1 \dots j$ from the boxes in which they currently reside and place them in the box we are considering, so now it contains the same set of items as the corresponding greedy box. In so doing, we clearly still have a feasible packing of items into boxes and since the number of boxes has not changed, this must still be an optimal solution; however, it now agrees with the greedy solution in one more box, contradicting the fact that we started with an optimal solution agreeing maximally with the greedy solution.

۲. [۳۰] جدول محاسبه Edit Distance زیر را برای تبدیل کلمه **brea** به **great** پر کنید. هزینه insert/delete/substitute همگی ۱ می‌باشد.

		g	r	e	a	t
	0	1	2	3	4	5
b	1	1	2	3	4	5
r	2	2	1	2	3	4
e	3	3	2	1	2	3
a	4	4	3	2	1	2

۳. [۲۰] دنباله اعداد و عملگرهای زیر را در نظر بگیرید.می‌دانیم از دنباله مقابل با انواع پرانتز گذاری (که ترتیب انجام عملیات ها را مشخص می‌کند) می‌توان مقادیر مختلفی را به دست آورد. با استفاده از (Dynamic programing) ماکسیمم این مقادیر را بدست بیاورید و روند فرایند را مرحله به مرحله بنویسید. عمل‌گرهای مجاز جمع، ضرب، تقسیم و بعلاوه هستند و کلیه اعداد صحیح، مثبت و بزرگتر از یک می‌باشند.

$$4 \times 3 - 6 \;/\; 2$$

1		2		3		4
۴	*	۳	-	۶	/	۲

	1	2	3	4
1	۴	۱۲	-۱۲	-۶
2		۳	-۳	-۱/۵
3			۶	۳
4				۲

Min

	1	2	3	4
1	۴	۱۲	۶	۹
2		۳	-۳	۰
3			۶	۳
4				۲

Max