Brückenbau

Zeitlimit: 3 s *Speicherlimit: 128 MB*

In einem breiten Fluss stehen n Säulen mit (nicht notwendigerweise) verschiedenen Höhen h_i. Diese sind der Reihe nach vom linken Ufer bis zum rechten angeordnet. Wir möchten auf diesen Säulen eine Brücke bauen. Um dies zu erreichen, wählen wir eine Teilmenge der Säulen aus und verbinden diese (in bestehender Reihenfolge) zu Teilabschnitten einer Brücke. Die erste und letzte Säule verwenden wir in jedem Fall.

Die Kosten zum Verbinden von Säulen i und j betragen $(h_i - h_j)^2$. Zusätzlich müssen unbenutzte Säulen entfernt werden, weil sie den Schiffsverkehr am Fluss behindern. Die Kosten, um die i-te Säule zu entfernen, betragen w_i. Dieser Wert kann auch negativ sein, denn einige beteiligte Kapitäne zahlen möglicherweise dafür, dass gewisse Säulen entfernt werden.

Was sind die minimal möglichen Kosten für den Bau einer Brücke, welche die erste und letzte Säule (nicht notwendigerweise direkt) miteinander verbindet?

Eingabe

Die 1. Zeile enthält die Anzahl der Säulen n.
Die 2. Zeile enthält in Reihenfolge die Höhen der Säulen h_i.
Die 3. Zeile enthält in Reihenfolge die Kosten der Säulen w_i.

Ausgabe

Gib die minimal möglichen Kosten für den Bau der Brücke aus.
Beachte, dass dieses Ergebnis negativ sein kann.

Limits

- $2 \leq n \leq 10^5$
- $0 \leq h_i \leq 10^6$
- $0 \leq |w_i| \leq 10^6$

Teilaufgabe 1 (30 Punkte)

- $n \leq 1\,000$

Teilaufgabe 2 (30 Punkte)

- Die optimale Lösung enthält höchstens 2 zusätzliche Säulen (neben der Ersten und Letzten).
- $|w_i| \leq 20$

Teilaufgabe 3 (40 Punkte)

- keine weiteren Einschränkungen

Beispiel

Eingabe

```
6
3 8 7 1 6 6
0 -1 9 1 2 0
```

Ausgabe

```
17
```

Palindromische Partitionen

Zeitlimit: 10 s *Speicherlimit: 128 MB*

Eine *Partition* einer Zeichenfolge s ist eine Folge von einer oder mehreren nicht-leeren Teilfolgen von s (wir nennen sie $a_1, a_2, a_3, \ldots, a_d$), sodass ihre Konkatenation s ist: $s = a_1 + a_2 + a_3 + \ldots + a_d$. Wir nennen diese Teilfolgen *“Stücke”*.

Wir können die Partition einer Zeichenfolge darstellen, indem wir jedes Stück in Klammern schreiben. Zum Beispiel kann die Zeichenfolge `"decode"` partitioniert werden als `(d)(ec)(ode)`, `(d)(e)(c)(od)(e)`, `(decod)(e)`, `(decode)`, `(de)(code)` oder auch auf einige andere Arten.

Eine Partition ist *palindromisch*, falls ihre Stücke ein Palindrom bilden, wenn wir jedes Stück als eine Einheit betrachten. Zum Beispiel, für die Zeichenfolge `"decode"` sind die einzigen palindromischen Partitionen `(de)(co)(de)` und `(decode)`. Dies illustriert gleichzeitig, dass jede Zeichenfolge eine triviale Partition mit genau einem Stück hat.

Deine Aufgabe ist es, die maximale Anzahl Stücke zu bestimmen, die eine palindromische Partition einer gegebenen Zeichenfolge haben kann.

Eingabe

Die erste Zeile der Eingabe enthält die Anzahl der Testfälle t. Die folgenden t Zeilen beschreiben die einzelnen Testfälle, welche aus jeweils einer einzelnen Zeichenfolge s bestehen, die nur kleine Buchstaben des englischen Alphabets enthält. Es gibt keine Leerzeichen in der Eingabe.

Ausgabe

Gib t Zeilen mit jeweils einer einzelnen Zahl aus: Die maximale Anzahl Stücke einer palindromischen Partition der jeweiligen Eingabezeichenfolge s.

Limits

Sei n die Länge der Eingabezeichenfolge s.

- $1 \leq t \leq 10$
- $1 \leq n \leq 10^6$

Teilaufgabe 1 (15 Punkte)

- $n \leq 30$

Teilaufgabe 2 (20 Punkte)

- $n \leq 300$

Teilaufgabe 3 (25 Punkte)

- $n \leq 10\,000$

Teilaufgabe 4 (40 Punkte)

- keine weiteren Einschränkungen

Beispiel

Eingabe	**Ausgabe**
`4`	`3`
`bonobo`	`5`
`deleted`	`7`
`racecar`	`1`
`racecars`	

Jagd

Zeitlimit: 4 s *Speicherlimit: 512 MB*

Die Katze Tom jagt wieder die Maus Jerry! Jerry versucht sich einen Vorteil zu verschaffen, indem er in einen Taubenschwarm läuft, wo es schwieriger für Tom ist, ihm zu folgen. Praktischerweise ist Jerry im Central Park in Ljubljana angekommen. Im Park gibt es n Statuen, welche von 1 bis n nummeriert sind, und $n-1$ überschneidungsfreie Passagen, die sie so verbinden, dass es möglich ist, jede Statue von jeder anderen Statue aus entlang der Passagen zu erreichen. Eng um jede Statue i gibt es p_i Tauben. Jerry hat v Brotkrümel in seiner Tasche. Wenn er einen Krümel bei der Statue, an der er sich aufhält, fallen lässt, werden die Tauben von allen benachbarten Statuen sofort zu dieser Statue hinfliegen um den Krümel zu essen. Dadurch ändert sich die aktuelle Anzahl der Tauben p an dieser und benachbarten Statuen.

Es passiert in der folgenden Reihenfolge: Zuerst kommt Jerry zur Statue i und trifft auf p_i Tauben. Dann lässt er einen Brotkrümel fallen und verlässt anschliessend die Statue. Zu guter Letzt fliegen die Tauben der benachbarten Statuen zur Statue i, bevor Jerry bei der nächsten Statue ankommt (Sie zählen also nicht zur Anzahl Tauben, die Jerry getroffen hat).

Jerry kann bei jeder Statue in den Park gelangen, einige Passagen entlang rennen (jedoch nie die gleiche Passage zwei mal benutzen) und dann den Park bei irgendeiner beliebigen Statue verlassen. Nachdem Jerry den Park verlassen hat, betritt ihn Tom und läuft genau die gleiche Route ab wie Jerry. Indem Jerry höchstens v Krümel fallen lässt, möchte er die Differenz zwischen der Anzahl der Tauben, die Tom entlang seiner Route antrifft, und der Anzahl der Tauben, die er selber antrifft, maximieren. Beachte, dass nur solche Tauben als angetroffen zählen, die sich bei einer Statue befinden, kurz bevor einer unserer Helden eintrifft. Beachte auch die Anmerkung zum Beispiel zur weiteren Erklärung.

Eingabe

Die erste Zeile enthält die Anzahl an Statuen n und die Anzahl an Brotkrümeln v. In der zweiten Zeile folgen n durch Leerzeichen getrennte Ganzzahlen $p_1 \dots p_n$. Die nächsten $n-1$ Zeilen beschreiben jeweils einen Pfad durch ein Nummernpaar a_i und b_i, das bedeutet, dass es einen Pfad zwischen den Statuen a_i und b_i gibt.

Ausgabe

Gib nur eine Ganzzahl aus: Die maximal mögliche Differenz zwischen der Anzahl an Tauben, auf die Tom trifft, und der, auf die Jerry trifft.

Limits

- $1 \leq n \leq 10^5$
- $0 \leq v \leq 100$
- $0 \leq p_i \leq 10^9$

Teilaufgabe 1 (20 Punkte)

- $1 \leq n \leq 10$

Teilaufgabe 2 (20 Punkte)

- $1 \leq n \leq 1000$

Teilaufgabe 3 (30 Punkte)

- Eine optimale Route startet bei Statue 1.

Teilaufgabe 4 (30 Punkte)

- keine weiteren Einschränkungen

Beispiel

Eingabe

```
12 2
2 3 3 8 1 5 6 7 8 3 5 4
2 1
2 7
3 4
4 7
7 6
5 6
6 8
6 9
7 10
10 11
10 12
```

Ausgabe

```
36
```

Anmerkung

Eine mögliche Lösung wäre: Zuerst betritt Jerry den Park bei Statue 6 und trifft dort 5 Tauben. Er hinterlässt einen Krümel, woraufhin $p_6 = 27$ und $p_5 = p_7 = p_8 = p_9 = 0$ gilt. Als nächstes begibt er sich zu Statue p_7, trifft auf 0 Tauben und hinterlässt erneut einen Krümel. Danach gilt $p_7 = 41$ und $p_2 = p_4 = p_6 = p_{10} = 0$. Nachdem er insgesamt $5 + 0 = 5$ Tauben getroffen hat, verlässt er den Park. Tom folgt ihm auf der selben Route und trifft $p_6 + p_7 = 0 + 41 = 41$ Tauben. Die Differenz ist $41 - 5 = 36$.