

دانشگاه پامین

- محدودیت زمان: 10 ثانیه
- محدودیت حافظه: 256 مگابایت
- برای حل این سوال، مجاز به استفاده از هیچ داده‌ساختار و تابع آماده‌ای نیستید.
- نمرات هر فاز با یک‌دیگر برابر است و نمرات قسمت‌های امتیازی متعاقباً اعلام می‌گردد.

در این متن، توضیحات کامل‌تر قسمت‌های پروژه نوشته شده است. لطفاً پیش از مطالعه‌ی این بخش، فایل داده‌شده پروژه را مطالعه کنید.

به دلیل مشکلات اینترنتی، سایت دانشگاه پامین دچار اختلال شده‌است. پس از پیگیری‌های متعدد دانشجویان، مسئولین دانشگاه تصمیم گرفتند که صورت مساله را پاک کرده و این سایت را از دسترس خارج کنند. این اتفاق، باعث مشکلات زیادی برای دانشجویان شده‌است؛ به همین دلیل، آن‌ها تصمیم گرفتند از دانشجویان دانشگاه دوست و همسایه، شریف، کمک بگیرند. از آن‌جایی که بهترین دانشکده‌ی این دانشگاه، دانشکده‌ی ریاضی می‌باشد(!!)، این عمل خطیر به عهده‌ی دانشجویان درس ساختمان‌داده این دانشکده واگذار شده‌است.

تعریف پروژه

در این پروژه ما به سه نوع داده کار خواهیم کرد:

- مشخصات دانشجویان (Student):
 - کد دانشجویی: یک عدد ۸ رقمی است. (هیچ دو دانشجویی دارای شماره دانشجویی یکسان نیستند)
 - نام دانشجو: یک رشته حرفی حداکثر ۳۰ کاراکتری است که فقط شامل حروف a..z و A..Z می‌تواند باشد.
- مشخصات دروس (Course):
 - کد درس: یک عدد دقیقاً ۵ رقمی است.

- نام درس: یک رشته حرفی حداکثر ۱۰ کاراکتری و فقط شامل حروف A..Z میتواند باشد .
- مشخصات نمرات (Grade):
- نیمسال اخذ: یک عدد ۵ رقمی است که ۴ رقم اول آن معرف سال و رقم آخر آن یکی از اعداد ۱ ، ۲ یا ۳ است.
- نمره: یک عدد بین صفر تا ۲۰ و با حداکثر یک رقم اعشار است.

فاز اول پروژه

هدف این فاز اضافه کردن، حذف کردن، ویرایش اطلاعات و سوالات مقدماتی درباره‌ی این اطلاعات می‌باشد. شما موظف هستید برای پیاده‌سازی این بخش از لیست استفاده کنید. در این فاز تعدادی عملیات به صورت زیر انجام می‌شود:

ADDS [student code] [student name]

این دستور، یک دانشجو با کد دانشجویی و اسم داده شده اضافه می‌کند. (پس ازین دستور نیازی به خروجی دادن چیزی نمی‌باشد)

ADDC [course code] [course name]

این دستور، یک درس با کد درس و اسم داده شده اضافه می‌کند. (پس ازین دستور نیازی به خروجی دادن چیزی نمی‌باشد)

ADDG [student code] [course code] [semester code] [grade]

این دستور، اطلاعات یک نمره را اضافه می‌کند که ورودی آن شامل کد دانشجویی فرد، کد درس، نیمسال اخذ و نمره‌ی آن فرد می‌باشد. تضمین می‌شود که هر فردی هر درس را حداکثر یک بار اخذ می‌کند و در همان نیمسال درس را پاس می‌کند؛ در واقع نمره‌ی صفر حداقل نمره‌ی پاس شدن درس می‌باشد. (پس ازین دستور نیازی به خروجی دادن چیزی نمی‌باشد)

EDITS [student code] [new student name]

این دستور، نام دانشجوی با کد داده‌شده را تغییر می‌دهد. تضمین می‌شود پیش ازین دستور، این دانشجو با این کد دانشجویی اضافه شده‌است. (پس ازین دستور نیازی به خروجی‌دادن چیزی نمی‌باشد)

EDITC [course code] [new course name]

این دستور، نام درس با کد داده‌شده را تغییر می‌دهد. تضمین می‌شود پیش ازین دستور، این درس با این کد اضافه شده‌است. (پس ازین دستور نیازی به خروجی‌دادن چیزی نمی‌باشد)

EDITG [student code] [course code] [new grade]

این دستور، نمره‌ی یک فرد در درس داده‌شده را تغییر می‌دهد. تضمین می‌شود که پیش از این دستور، این فرد، این درس را گذرانده است. (پس ازین دستور نیازی به خروجی‌دادن چیزی نمی‌باشد)

DELETES [student code]

این دستور، دانشجوی با کد داده‌شده را حذف می‌کند. تضمین می‌شود پیش ازین دستور، این دانشجو با این کد دانشجویی اضافه شده‌است. (پس ازین دستور نیازی به خروجی‌دادن چیزی نمی‌باشد)

DELETEC [course code]

این دستور، درس با کد داده‌شده را حذف می‌کند. تضمین می‌شود پیش ازین دستور، این درس با این کد اضافه شده‌است. (پس ازین دستور نیازی به خروجی‌دادن چیزی نمی‌باشد)

DELETEG [student code] [course code]

این دستور، نمره‌ی دانشجوی داده‌شده در درس را حذف می‌کند. تضمین می‌شود که پیش از این دستور، این فرد، این درس را گذرانده است. (پس ازین دستور نیازی به خروجی‌دادن چیزی نمی‌باشد)

NUMBERC [student code]

با گرفتن این دستور، باید تعداد دروس اخذ شده توسط دانشجوی با کد داده‌شده را خروجی دهید.

NUMBERS [course code]

با گرفتن این دستور، باید تعداد دانشجویانی که این درس با کد داده‌شده را اخذ کرده‌اند را خروجی دهید.

فاز دوم پروژه

هدف ازین فاز، ایجاد امکان جست‌وجوی دانشجویها و دروس می‌باشد. دقت کنید که برای زدن این بخش، پیاده‌سازی فاز اول نیاز است و دستورات فاز اول در این فاز نیز اعمال خواهد شد. بدین‌منظور شما باید عملیات‌های زیر را انجام دهید:

SEARCHSN [student name]

با گرفتن این دستور، باید اطلاعات کامل دانشجو را به صورت زیر خروجی دهید:

[student code] [student name] [number of taken courses]

[course code 1] [semester 1] [grade 1]

[course code 2] [semester 2] [grade 2]

.

.

.

[course code k] [semester k] [grade k]

در واقع در خط اول، کد دانشجویی، نام دانشجو و تعداد دروس اخذ شده‌ی او را خروجی دهید و سپس به تعداد دروس اخذ شده‌ی او، در هر خط اطلاعات یک درس شامل کد درس، نیمسال اخذ آن درس توسط دانشجو و نمره‌ی دانشجو در این درس را خروجی بدهید. ترتیب دروس باید مانند ترتیب اضافه‌شدن آن‌ها در دستورات باشد.

SEARCHCN [course name]

با گرفتن این دستور، باید اطلاعات کامل درس را به صورت زیر خروجی دهید:

[course code] [course name] [number of students taken this course]

[student code 1] [semester 1] [grade 1]

[student code 2] [semester 2] [grade 2]

.

.

.

[student code k] [semester k] [grade k]

در واقع در خط اول، کد درس، نام درس و تعداد دانشجویانی که این درس را اخذ کرده‌اند را خروجی دهید و سپس به تعداد این دانشجویان، در هر خط اطلاعات یک دانشجو شامل کد دانشجو، نیمسال اخذ آن درس توسط دانشجو و نمره‌ی دانشجو در این درس را خروجی بدهید. ترتیب دانشجوها باید مانند ترتیب اضافه‌شدن آن‌ها در دستورات باشد.

SEARCHSC [student code]

با گرفتن این دستور، باید اطلاعات کامل دانشجو را به صورت زیر خروجی دهید:

[hash]

[student code] [student name] [number of taken courses]

[course code 1] [semester 1] [grade 1]

[course code 2] [semester 2] [grade 2]

.

.
.
[course code k] [semester k] [grade k]

در واقع در خط اول مقدار درهم‌سازی شده‌ی کد دانشجوی داده‌شده و در خط دوم، کد دانشجویی، نام دانشجو و تعداد دروس اخذ شده‌ی او را خروجی دهید و سپس به تعداد دروس اخذ شده‌ی او، در هر خط اطلاعات یک درس شامل کد درس، نیمسال اخذ آن درس توسط دانشجو و نمره‌ی دانشجو در این درس را خروجی بدهید. ترتیب دروس باید مانند ترتیب اضافه‌شدن آن‌ها در دستورات باشد.

SEARCHCC [course code]

با گرفتن این دستور، باید اطلاعات کامل درس را به صورت زیر خروجی دهید:

[hash]
[course code] [course name] [number of students taken this course]
[student code 1] [semester 1] [grade 1]
[student code 2] [semester 2] [grade 2]
.
.
.
[student code k] [semester k] [grade k]

در واقع در خط اول مقدار درهم‌سازی شده‌ی کد درس داده‌شده و در خط دوم، کد درس، نام درس و تعداد دانشجویانی که این درس را اخذ کرده‌اند را خروجی دهید و سپس به تعداد این دانشجویان، در هر خط اطلاعات یک دانشجو شامل کد دانشجو، نیمسال اخذ آن درس توسط دانشجو و نمره‌ی دانشجو در این درس را خروجی بدهید. ترتیب دانشجوها باید مانند ترتیب اضافه‌شدن آن‌ها در دستورات باشد.

نکات فاز دوم پروژه

۱. ممکن است بین دستورات جست‌وجو، دستورات حذف و اضافه و ویرایش نیز داده شوند.
۲. برای جست‌وجو با اسم درس و اسم دانشجو باید حتماً از درخت جست‌وجوی دودویی و برای جست‌وجو با کد درس و کد دانشجو باید حتماً از جدول درهم‌سازی استفاده کنید.

۳. برای پیاده‌سازی درخت جست‌وجوی دودویی مجاز هستید از هر نوع درخت جست‌وجوی دودویی استفاده کنید ولی برای مقایسه‌ی رشته‌ها باید از ترتیب الفبایی استفاده کنید.

۴. برای پیاده‌سازی جدول درهم‌سازی حتماً باید از روش درهم‌سازی پویای خانوادگی سراسری توابع درهم‌سازی زیر استفاده کنید. (خانوادگی سراسری توابع درهم‌سازی معرفی‌شده در کتاب)

$$h_{a,b}(k) = ((ak + b) \bmod p) \bmod m$$

$$H_{p,m} = \{h_{a,b} : a \in Z_p^* \text{ and } b \in Z_p\}$$

در ابتدای ورودی با دادن a, b و p تابع درهم‌سازی مشخص می‌شود.

فاز سوم پروژه

در این فاز شما باید تعدادی عملیات تکمیلی را پیاده‌سازی کنید. تضمین می‌شود در بین دستورات فاز سوم، حذف و اضافه و ویرایش اطلاعات رخ نمی‌دهد. این فاز شامل چند بخش می‌باشد:

ارتباط دروس:

دو درس A و B مرتبط هستند اگر و تنها اگر یک دنباله از دروس مثل

$$C_0 = A, C_1, C_2, \dots, C_t = B$$

وجود داشته باشد که C_i و C_{i+1} به ازای هر $0 \leq i < t$ مرتبط مستقیم باشند.

دو درس A و B مرتبط مستقیم هستند اگر و تنها اگر تعداد دانشجویان مشترک دو درس بیش از نصف هر درس را تشکیل دهند.

دستورات این بخش به صورت زیر می‌باشد:

ISRELATIVE [course code 1] [course code 2]

شما باید با گرفتن این دستور بررسی کنید که آیا این دو درس به هم مرتبط هستند یا خیر؛ سپس "yes" یا "no" خروجی بدهید.

ALLRELATIVE [course code]

شما باید با گرفتن این دستور در یک خط، کد دروس مرتبط با این درس را خروجی بدهید. ترتیب خروجی دادن این دروس، باید به ترتیب مرتب‌شده از کوچک به بزرگ باشد.

ریسک در اخذ دروس:

میزان مطلوب بودن اخذ یک درس در یک نیمسال برابر با میانگین نمرات دانشجویان در آن درس در آن نیمسال است.

این بخش تنها یک دستور دارد که به صورت زیر است:

MINRISK [student code]

شما باید با گرفتن این دستور، ترتیب بهینه‌ی اخذ دروس را به صورت زیر خروجی دهید:

```
[semester 1] [course code 1, 1] [course code 1, 2] ... [course code 1, k_1]
[semester 2] [course code 2, 1] [course code 2, 2] ... [course code 2, k_2]
.
.
.
[semester t] [course code t, 1] [course code t, 2] ... [course code t, k_t]
```

در خروجی، باید شرایط زیر صدق کند:

۱. فقط هر درسی که دانشجو اخذ کرده‌است، دقیقاً یک‌بار در خروجی تکرار شود
۲. در هر نیمسال، باید تعداد درس‌های اخذ شده پیشنهاد شده، برابر با تعداد درس‌های اخذ شده‌ی در همان نیمسال باشد.

۳. خروجی مورد نظر، بین تمام خروجی‌های معتبر باید بیش‌ترین مجموع میزان مطلوبیت تمام دروس را داشته‌باشد.

۴. ترتیب نیم‌سال‌ها باید به ترتیب مرتب از کوچک به بزرگ خروجی داده شود.

۵. در هر نیم‌سال، ترتیب دروس باید به ترتیب مرتب کد درس از کوچک به بزرگ خروجی داده شود.

رتبه‌ی دانشجوها:

دانشجوی A از دانشجوی B رتبه‌ی بهتری دارد اگر و تنها اگر یک دنباله از دانشجوها مثل

$$C_0 = A, C_1, C_2, \dots, C_t = B$$

وجود داشته باشد که به ازای هر $0 \leq i < t$ ، دانشجوی C_i به طور مستقیم رتبه‌ی بهتری از دانشجوی C_{i+1} داشته باشد و دنباله‌ای از دانشجوها مثل

$$C'_0 = B, C'_1, C'_2, \dots, C'_{t'} = A$$

وجود نداشته‌باشد که به ازای هر $0 \leq i < t'$ ، دانشجوی C'_i به طور مستقیم رتبه‌ی بهتری از دانشجوی C'_{i+1} داشته باشد.

دانشجوی A از دانشجوی B به طور مستقیم رتبه‌ی بهتری دارد اگر و تنها اگر در بیش از نصف دروس مشترک آن‌ها، دانشجوی A نمره‌ی اکیدا بالاتری کسب کرده باشد. دو دانشجو که دارای درس مشترک نیستند، هیچ‌کدام به طور مستقیم رتبه‌ی بهتری از دیگری ندارد. تنها دستور این بخش به صورت زیر است:

COMPARE [student code 1] [student code 2]

اگر دانشجوی اول از دانشجوی دوم، رتبه‌ی بهتری داشته‌باشد ">"، در صورتی که دانشجوی دوم، رتبه‌ی بهتری از دانشجوی اول داشته باشد، "<" و در صورتی که این هیچ‌کدام رتبه‌ی بهتری از دیگری نداشته‌باشد، "?" چاپ کنید.

قسمت‌های امتیازی

۱. در فاز اول، در بخش حذف‌کردن دانشجو، در صورتی که مساله را با پیچیدگی زمانی $O(T)$ حل کنید که T تعداد دروس اخذ شده توسط دانشجو باشد، نمره‌ی امتیازی دریافت خواهید کرد. (به طور مشابه باید برای حذف کردن درس، باید این تغییرات اعمال شود).
۲. در فاز دوم، در بخش جست‌وجو با درخت جست‌وجوی دودویی اگر درخت شما به صورت متوازن باشد، نمره‌ی امتیازی دریافت خواهید کرد.
۳. در فاز سوم، در بخش انتخاب کردن ترتیب بهینه‌ی اخذ دروس، در صورتی که الگوریتم شما به صورت چندجمله‌ای باشد (دقت کنید الگوریتم شما باید با محدودیت‌های سوال نیز سازگار باشد)، نمره‌ی امتیازی دریافت خواهید کرد.

ورودی

در خط اول T, a, b و p داده می‌شود که به ترتیب تعداد دستورها و مقادیر لازم برای ساخت جدول درهم‌سازی می‌باشند. در T خط بعدی در هر خط یک دستور داده می‌شود. توضیحات تست‌کیس‌های هر بخش در جدول زیر آمده‌است:

نمره	تست کیس ها	محدودیت ها	توضیحات	فاز
30	1-5	$n \leq 5 * 10^5$	دستورات NUMBER , ADD	1
		$m \leq 5 * 10^3$		
		$n * g, m * g \leq 10^8$		
30	6-10	$n \leq 5 * 10^5$	دستورات NUMBER , EDIT , ADD	1
		$m \leq 5 * 10^3$		
		$n * g, m * g \leq 10^8$		

نمره	تست کیس ها	محدودیت ها	توضیحات	فاز
40	11-15	$n \leq 5 * 10^3$ $m \leq 5 * 10^3$ $g \leq 10^4$	دستورات NUMBER , DELETE , ADD	1
25~50	تحویل	-	بخش امتیازی فاز اول شامل تمام دستورات	1
25	16-22	$n \leq 5 * 10^3$ $T \leq 2 * 10^5$	دستورات فاز اول و SEARCHSN	2
25	23-30	$m \leq 5 * 10^3$ $T \leq 2 * 10^5$	دستورات فاز اول و SEARCHCN	2
25	31-35	$n \leq 5 * 10^3$ $T \leq 2 * 10^5$ $a, b, p \leq 2 * 10^9$	دستورات فاز اول و SEARCHSC	2

نمره	تست کیس ها	محدودیت ها	توضیحات	فاز
25	35-40	$n \leq 5 * 10^3$ $T \leq 2 * 10^5$ $a, b, p \leq 2 * 10^9$	دستورات فاز اول و SEARCHCC	2
25~50	تحویل	-	بخش امتیازی فاز دوم شامل تمام دستورات SEARCHSN , SEARCHCN و فاز اول	2
20	41-45	$n, m \leq 50$ $g \leq 2000$ $T \leq 1 * 10^5$	دستورات فاز اول و ISRELATED	3
20	46-50	$n, m \leq 50$ $g \leq 2000$ $T \leq 10^5$	دستورات فاز اول و ALLRELATED	3
30	51-55	$n, m \leq 10$ $g \leq 100$ $MINRISK \leq 4$	دستورات فاز اول و MINRISK	3

نمره	تست کیس ها	محدودیت ها	توضیحات	فاز
		$n, m \leq 50$		
30	56-60	$g \leq 500$	دستورات فاز اول و COMPARE	3
		$T \leq 5 * 10^4$		
25~50	تحویل	-	بخش امتیازی فاز سوم شامل تمام دستورات فاز اول و MINRISK	3

اطلاعات این جدول در ادامه تکمیل می‌گردد.

n = تعداد دانشجو ها

m = تعداد درس ها

g = تعداد تمام نمرات

خروجی

شما باید برای هر دستور که نیاز به خروجی دارد، مطابق با فرمت ارائه‌شده، خروجی دهید.

ورودی نمونه 1

```

22 1 1 1
ADDS 11111111 A
ADDS 22222222 B
ADDS 33333333 C
ADDC 11111 W
ADDC 22222 X
ADDC 33333 Y
ADDG 11111111 11111 10001 11

```

ADDG 11111111 22222 10002 12.1
 ADDG 22222222 22222 10003 20.0
 ADDG 33333333 33333 10003 14
 NUMBERC 11111111
 NUMBERS 11111
 EDITS 11111111 Q
 EDITC 11111 P
 EDITG 11111111 22222 14.7
 NUMBERS 11111
 DELETEDC 22222
 DELETES 11111111
 DELETEDG 33333333 33333
 NUMBERC 22222222
 NUMBERS 11111
 NUMBERS 33333

خروجی نمونه 1

2
 1
 1
 0
 0
 0

این تست مربوط به فاز اول است.

ورودی نمونه 2

15 3 5 1000000007
 ADDS 40528315 URU
 ADDS 28839949 YZX
 ADDS 55810244 DVL
 ADDC 17926 ALD
 ADDC 95218 LXB
 ADDC 21211 UGQ

ADDG 40528315 17926 10002 19.1
ADDG 28839949 17926 10001 17.4
SEARCHSN URU
SEARCHCN ALD
EDITS 40528315 A
EDITC 17926 Z
SEARCHSN A
SEARCHCN Z
DELETEC 17926

خروجی نمونه 2

40528315 URU 1
17926 10002 19.1
17926 ALD 2
40528315 10002 19.1
28839949 10001 17.4
40528315 A 1
17926 10002 19.1
17926 Z 2
40528315 10002 19.1
28839949 10001 17.4

این تست مربوط به فاز دوم بخش درخت دودویی است.

ورودی نمونه 3

23 3 5 1000000007
ADDS 40528315 URU
ADDS 28839949 YZX
ADDS 55810244 DVL
ADDS 12312342 JIE
ADDS 32452152 PRE
ADDC 17926 ALD
ADDC 95218 LXB
ADDC 21211 UGQ

ADDG 40528315 17926 10002 19.1
ADDG 28839949 17926 10001 17.4
SEARCHSC 40528315
SEARCHCC 17926
EDITS 40528315 A
EDITC 17926 Z
SEARCHSC 40528315
SEARCHCC 17926
DELETEDC 17926
SEARCHCC 95218
DELETES 32452152
DELETES 12312342
DELETES 55810244
DELETES 28839949
SEARCHSC 40528315

خروجی نمونه 3

6
40528315 URU 1
17926 10002 19.1
3
17926 ALD 2
40528315 10002 19.1
28839949 10001 17.4
6
40528315 A 1
17926 10002 19.1
3
17926 Z 2
40528315 10002 19.1
28839949 10001 17.4
3
95218 LXB 0
2
40528315 A 0

این تست مربوط به فاز دوم بخش جدول درهم سازی پویا است.

ورودی نمونه 4

```
33 1 1 1
ADDS 11111111 A
ADDS 22222222 B
ADDS 33333333 C
ADDS 44444444 D
ADDC 10003 X
ADDC 10002 Y
ADDC 10001 Z
ADDC 10007 T
ADDC 10005 W
ADDC 10006 P
ADDC 10004 Q
ADDC 10008 R
ADDG 11111111 10003 10001 20
ADDG 11111111 10002 10001 19.1
ADDG 11111111 10001 10003 3.3
ADDG 11111111 10007 10002 10.3
ADDG 22222222 10003 20001 18.3
ADDG 22222222 10002 20001 4
ADDG 22222222 10005 20001 5.5
ADDG 22222222 10006 20003 10
ADDG 33333333 10003 34441 19.9
ADDG 33333333 10001 32641 1.7
ADDG 33333333 10006 32241 16.5
ADDG 33333333 10004 32741 0
EDITC 10008 O
DELETES 44444444
ISRELATIVE 10001 10003
ISRELATIVE 10002 10005
ISRELATIVE 10008 10006
ISRELATIVE 10006 10005
ALLRELATIVE 10008
```

ALLRELATIVE 10005

ALLRELATIVE 10003

خروجی نمونه 4

yes

no

no

no

10001 10002 10006

این تست مربوط به فاز سوم بخش ارتباط دروس است.

ورودی نمونه 5

30 1 1 1

ADDS 11111111 A

ADDS 22222222 B

ADDS 33333333 C

ADDS 44444444 D

ADDC 10003 X

ADDC 10002 Y

ADDC 10001 Z

ADDC 10007 T

ADDC 10005 W

ADDC 10006 P

ADDC 10004 Q

ADDC 10008 R

ADDG 11111111 10003 10001 20

ADDG 11111111 10002 10001 19.1

ADDG 11111111 10001 10003 3.3

ADDG 11111111 10007 10002 10.3

ADDG 22222222 10003 20001 18.3

ADDG 22222222 10002 20001 19.8

```

ADDG 22222222 10005 20001 5.5
ADDG 22222222 10006 20003 10
ADDG 33333333 10003 34441 19.9
ADDG 33333333 10001 32641 1.7
ADDG 33333333 10006 32241 16.5
ADDG 33333333 10004 32741 0
ADDG 44444444 10003 43421 18.3
DELETEDG 11111111 10003
EDITG 11111111 10002 1.1
COMPARE 11111111 22222222
COMPARE 22222222 33333333
COMPARE 11111111 44444444

```

خروجی نمونه 5

```

?
?
>

```

این تست مربوط به فاز سوم بخش رتبه ی دانشجو است.

ورودی نمونه 6

```

30 1 1 1
ADDS 11111111 A
ADDS 22222222 B
ADDS 33333333 C
ADDS 44444444 D
ADDC 33333 X
ADDC 22222 Y
ADDC 11111 Z
ADDC 77777 T
ADDC 55555 W
ADDC 66666 P
ADDC 44444 Q
ADDC 88888 R

```

```

ADDG 11111111 33333 10001 20
ADDG 11111111 22222 10001 19.1
ADDG 11111111 11111 10002 3.3
ADDG 11111111 77777 10002 10.3
ADDG 22222222 33333 10002 18.3
ADDG 22222222 22222 10001 19.8
ADDG 22222222 55555 10001 5.5
ADDG 22222222 66666 10002 10
ADDG 33333333 33333 10001 19.9
ADDG 33333333 11111 10002 1.7
ADDG 33333333 66666 10001 16.5
ADDG 33333333 44444 10002 0
ADDG 44444444 33333 10001 18.3
DELETES 44444444
EDITG 11111111 22222 1.1
MINRISK 11111111
MINRISK 22222222
MINRISK 33333333

```

خروجی نمونه 6

```

10001 22222 33333
10002 11111 77777
10001 22222 66666
10002 33333 55555
10001 33333 66666
10002 11111 44444

```

این تست مربوط به فاز سوم بخش ریسک اخذ درس است.