

### Zadatak 1 – Tobogan

Mirko i Slavko su u posjeti lunaparku. Najveća atrakcija u lunaparku je vodeni tobogan. Nažalost, red u kojem se čeka na ulaz u vodeni tobogan je uvijek pun. Sljedeća slika prikazuje primjer tlocrta reda u kojem se čeka 12 minuta. Početak reda je u polju 1. Mirko je ušao u red; sva mjesta ispred njega su puna. Nakon jednog minuta, osoba na kraju reda ulazi na tobogan, svi u redu se pomiču unaprijed jedno mjesto i nova osoba ulazi u red. Slavko je prije ulaska u red otišao da kupi sladoled. Tačno K minuta nakon Mirka, Slavko uspijeva ući u red. Mirko i Slavko mogu čavrljati ako se nalaze na poljima **susjednima u nekom od osam smjerova** (gore, dolje, lijevo, desno i četiri dijagonalna smjera). Koliko minuta će provesti u čavrljanju? U gornjem primjeru, ako Slavko uđe u red dva minuta nakon Mirka, moći će čavrljati ukupno tri minuta dok čekaju u redu: dok je Mirko u polju 6, a Slavko u polju 4; dok je Mirko u polju 7, a Slavko u polju 5; dok je Mirko u polju 11, a Slavko u polju 9. **Ulaz:** Prvi red sadrži dužinu reda  $L$  ( $2 \leq L \leq 250$ ) i opis tlocrta reda, niz od  $L-1$  velikih slova. Niz opisuje put koji pređe svaki gost parka koji čeka u redu. Pojavljivaće se samo slova 'L', 'R', 'U' i 'D', a označavaju pomicanje na polje lijevo, desno, gore i dolje od trenutnog. Drugi red sadrži prirodni broj  $K$  ( $1 \leq K < L$ ), koliko minuta nakon Mirka je Slavko ušao u red. Niz slova u prvom redu ulaza predstavljaće ispravan tlocrt reda, tj. red neće sjeći sam sebe. **Izlaz:** Štampati koliko će minuta Mirko i Slavko biti dovoljno blizu da čavrljaju.

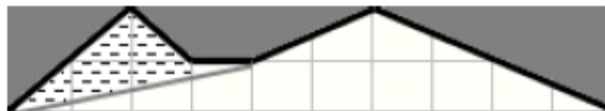
1	2	3	4	5
10	9	8	7	6
11				
12				

### Zadatak 2 – Obala

Marko, sekretar za turizam jedne primorske opštine je odlučio ograditi dio obale kako bi ga mogao zaliti betonom. Kako bi turisti što više “uživali”, on želi betonirati dio obale **što je moguće veće površine**.

Linija obale data je izlomljenom linijom koja ne presijeca samu sebe; linija je opisana nizom tjemena  $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$ ,  $x_1 < x_2 < \dots < x_N$ .

More se nalazi **iznad**, a obala **ispod** izlomljene linije. Marko ima konopac dužine  $L$  metara. On će **odabrati dva tjemena** izlomljene linije i u njihovim zabiti dva štapa. Potom će razapeti konopac (ili samo dio konopca) između ta dva štapa tako da



konopac bude napet. Duž koju čini konopac **ne smije sjeći more**, ali je dopušteno da **dodiruje liniju obale**.

Nakon razapinjanja konopca, Marko će betonirati dio obale između konopca i mora. Slika odgovara trećem primjeru. Crna izlomljena linija predstavlja liniju obale, more je zasivljeno. Duž koja predstavlja razapeti konopac predstavljen je sivom bojom. Betonirani dio obale je osjenčen. Napišite program koji za datu liniju obale i dužinu konopca, odrediti kolika je **maksimalna površina obale** koju Marko može ograditi i zatim izbetonirati.

Primjetite da tražena površina može biti i nula. **Ulaz:** U prvom redu ulaza nalaze se cijeli brojevi  $N$  i  $L$ ,  $3 \leq N \leq 5\,000$ ,  $0 \leq L \leq 1\,000\,000$ . Broj  $N$  predstavlja broj tjemena izlomljene linije kojom je opisana linija obale. Broj  $L$  predstavlja dužinu Markovog konopca. U svakom od sljedećih  $N$  redova nalaze se po dva cijela broja  $x_i$  i  $y_i$  odvojena blankom,  $0 \leq x_i, y_i \leq 1\,000\,000$  – koordinate tjemena linije obale.

**Izlaz:** U jedinom redu izlaza treba štampati jedan realni broj – najveću moguću površinu obale koju Marko može ograditi i betonirati. Rješenje štampati sa tačno jednom decimalom. Možete pretpostaviti da će površina uvijek biti ili cijeli broj ili polovina cijelog broja.

### Zadatak 3 – Čuvar

Poznata kompanija je vlasnik poslovne zgrade u centru grada. Krov zgrade je ravan i ima oblik kvadrata. Na krovu se nalazi nekoliko otvora koji su se pokazali kao veliki sigurnosni problem. Naime, u posljednjih mjesec dana dogodilo se nekoliko provala u zgradu, a provalnici su provaljivali upravo kroz otvore na krovu. Stoga je kompanija odlučila na krov postaviti psa čuvara koji će paziti da neko ne bi pokušao ući kroz otvore. Međutim, lokalni pet-shop im je podvalio vrlo nespretnog psa koji je već prvi dana zamalo pao sa zgrade. Zbog toga su odlučili staviti psa na lanac i to tako da:

- lanac je učvršćen u tački sa cjelobrojnim koordinatama (ako je krov kvadrat dimenzija  $10 \times 10$  metara, onda njegov jugozapadni vrh ima koordinate  $(0, 0)$ , a sjeveroistočni vrh ima oordinate  $(10, 10)$ )
- pas ne može pasti sa zgrade (ali smije doći do same ivicije)
- pas može doći do centra svakog od otvora
- lanac ne može biti učvršćen u tački koja je središte nekog otvora

Odredite koordinate tačke na krovu u kojoj lanac može biti učvršćen.

**Ulaz:** U prvom redu ulaza je prirodni broj  $N \leq 100$ , broj test primjera koje će sadržati ulazni podaci.

Slijede opisi  $N$  test primjera. Svaki opis započinje redom u kojem se nalaze dva cijela broja  $S$  i  $H$ , odvojena jednim blankom. Broj  $S$  je paran,  $2 \leq S \leq 40$ ,  $1 \leq H \leq 50$ . Broj  $S$  predstavlja dužinu stranice krova u metrima, a  $H$  broj otvora na krovu. Sljedećih  $H$  redova sadrže po dva cijela broja  $X$  i  $Y$ , koordinate centra odgovarajućeg otvora. Centri otvora će uvijek biti u unutrašnjosti krova (neće biti niti na

njegovoj ivici). Dva otvora se neće preklapati. **Izlaz:** Za svaki test primjer potrebno je štampati jedan red koji sadrži koordinate X i Y (odvojene jednim razmakom). Te koordinate predstavljaju tačku u kojoj treba učvrstiti lanac tako da budu zadovoljeni uslovi zadatka. Ako postoji više takvih tačaka, štampati onu koja ima najmanju X-koordinatu; ako je i dalje preostalo više tačaka, od njih štampati onu koja ima najmanju Y-koordinatu. Ako tražena tačka ne postoji, štampati red "-1 -1" kao odgovor na taj test primjer.

Zadatak 1		Zadatak 2			Zadatak 3	
<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>	<b>Izlaz</b>
12 RRRRDLLLLDD	9 RDRUURDR	5 4	3 10	6 5	3	3 6
2	3	0 0	100 100	0 0	10 2	-1 -1
<b>Izlaz</b>	<b>Izlaz</b>	1 3	101 0	2 2	6 6	2 2
3	2	2 0	102 100	3 1	5 4	
		3 3	<b>Izlaz</b>	4 1	20 2	
		4 0	0.0	6 2	1 1	
		<b>Izlaz</b>		10 0	19 19	
		6.0		<b>Izlaz</b>	10 3	
				2.5	1 1	
					1 2	
					1 3	

#### Zadatak 4 – Djelioci

Marija je dobila zadatak da izračuna sumu svih djelilaca broja 100. Tačan rezultat je, naravno,  $f(100) = 1 + 2 + 4 + 5 + 10 + 20 + 25 + 50 + 100 = 217$ . Nakon što je taj zadatak riješila "kao od šale", dobila je zadatak da sazna zbir suma djelilaca svih prirodnih brojeva u intervalu od L do R, uključivo. No kako Marija zbog drugih obaveza nema vremena da provede sate u računanju, vaš zadatak je da to izračunate umjesto nje. **Ulaz:** U prvom i jedinom redu nalaze se prirodni brojevi L i R takvi da  $1 \leq L \leq R \leq 1\,000\,000$ . **Izlaz:** U prvi i jedini red izlaza štampati traženu sumu  $f(L) + f(L+1) + \dots + f(R-1) + f(R)$ .

#### Zadatak 5 – Programer

Mirko je s jednim klijentom dogovorio izradu velikog informacionog sistema, a danas je taj veliki dan kada svi dogovoreni programi moraju biti isporučeni. Međutim, većina programa nije završena. Mirko se zbog toga pomalo uznemirio i sada ponovo čita ugovor. Kazna zbog prekoračenja roka isporuke za svaki program računa posebno i to pomoću kubnog polinoma  $at^3+bt^2+ct+d$ , gdje je t broj dana kašnjenja (u trenutku predaje programa), dok su a, b, c i d koeficijenti koji su specifični za svaki pojedini program. Na Mirkovu sreću, **ukupna kazna** koju treba platiti nije jednaka zbiru svih kazni, nego samo **najvećoj pojedinačnoj kazni**. Kao što ste i mogli naslutiti, samo Mirko piše programe i to na jednom računaru, a za svaki program je poznat tačan broj dana da bi ga Mirko završio i predao. Broj dana je jednak nula za programe koji su završeni, ali ne i predati klijentu. Napišite program koji će odrediti koliki je **najmanji iznos kazne** koju će Mirko platiti ako napravi optimalni poredak isporuke programa. **Ulaz:** U prvom redu se nalazi prirodni broj N,  $1 \leq N \leq 2\,000$ , broj programa. U svakom od sljedećih N redova nalazi se po pet nenegativnih cijelih brojeva manjih ili jednakih od 100. Ti brojevi predstavljaju broj dana koji je potreban da se određeni program završi i preda, te nakon njega koeficijenti a, b, c i d specifični za taj program. **Izlaz:** U prvi i jedini red štampati traženu minimalnu kaznu koju Mirko može dobiti. **Napomena:** koristite 64-bitni cjelobrojni tip podataka (int64 u Pascalu, long long u C/C++).

**Pojašnjenje 1. test primjera:** Ako odmah počne rad na datom programu, može ga završiti za 5 dana pa ga isporučiti. Kazna nam je tada jednaka  $0*5^3+3*5^2+2*5+1=86$ .

**Pojašnjenje 2. test primjera:** Ako prvo napravi 1. program, a potom 2. program, kazna će biti  $\max\{2*5+3, 1*(5+2)+2\}=13$ . Ako zamijeni redosljed, tada je kazna jednaka  $\max\{1*2+2, 2*(2+5)+3\}=17$ . Dakle, minimalna kazna je 13.

Zadatak 4			Zadatak 5		
<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>	<b>Ulaz</b>
100 100	1 10	7 9	1	2	5
<b>Izlaz</b>	<b>Izlaz</b>	<b>Izlaz</b>	5 0 3 2 1	5 0 0 2 3	87 0 0 0 16
217	87	36	<b>Izlaz</b>	2 0 0 1 2	97 74 94 0 25
			86	<b>Izlaz</b>	67 83 5 94 60
				13	41 3 78 81 13
					28 0 10 88 82
					<b>Izlaz</b>
					328938105