ZADANIE 1 – SUMMA SUMMARRUM, TO SUMA SUM

Zadanie zaproponował: dr inż. Mariusz Pleszczyński, Wydział Matematyki Stosowanej, Politechnika Śląska

Oznaczmy, że dla dowolnego $k \in \mathbb{N}$

$$\sum_{i=1}^{k} \frac{1}{i} = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{k} = s(k).$$

Oblicz

$$\left(\frac{4-1}{2+1} + \frac{9-4}{3+2} + \frac{16-9}{4+3} + \cdots + \frac{4084441-4080400}{2021+2020}\right) + \left(\frac{4-1}{1} + \frac{9-4}{2} + \frac{16-9}{3} + \cdots + \frac{4084441-4080400}{2020}\right) + \left(\frac{1}{2\cdot 1} + \frac{1}{3\cdot 2} + \frac{1}{4\cdot 3} + \cdots + \frac{1}{2021\cdot 2020}\right).$$

Odpowiedź podaj w postaci sumy: $r + \frac{p}{q} + s(t)$, gdzie $r, p, q, t \in \mathbb{N}$, $p < q$.

Napisz program, który będzie zwracał poszukiwaną sumę w postaci podanej sumy, jeśli jako argument tego programu podamy liczbę $n \in \mathbb{N}$, będącą ilością składników każdej składowej obliczanej sumy (w pierwszej części zadania $n = 2020$).

Przykładowo, jeśli podamy $n = 3$, to obliczana suma miałaby postać

$$\left(\frac{4-1}{2+1} + \frac{9-4}{3+2} + \frac{16-9}{4+3}\right) + \left(\frac{4-1}{1} + \frac{9-4}{2} + \frac{16-9}{3}\right) + \left(\frac{1}{2\cdot 1} + \frac{1}{3\cdot 2} + \frac{1}{4\cdot 3}\right),$$

a program zwróciłby napis: $9 + \frac{3}{4} + s(3)$.

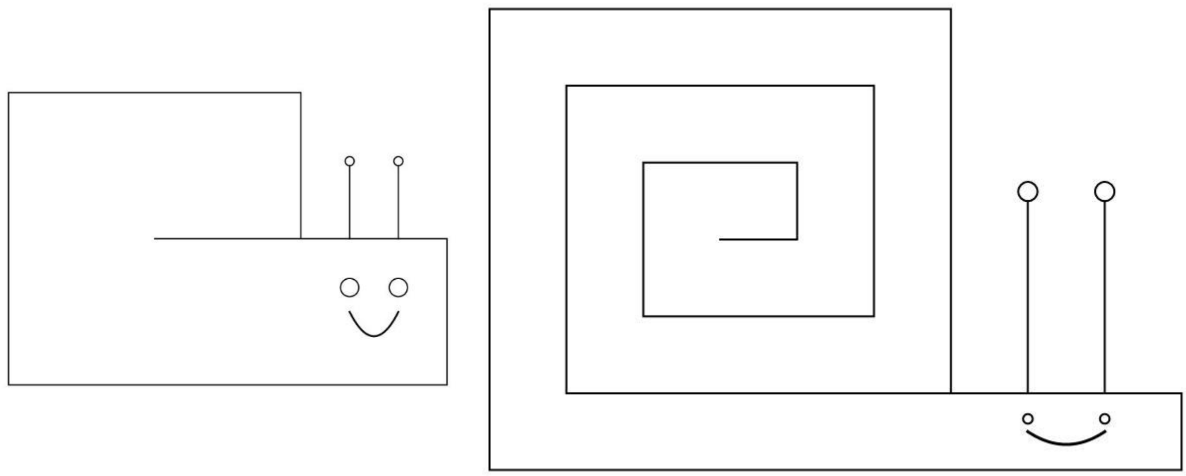
Uwaga: to zadanie w 95% jest matematyczne, więc komplet punktów można otrzymać za wyprowadzenie wzoru na tę sumę, a program korzystał będzie z tego wzoru.
Uwaga: postać $r + \frac{p}{q} + s(t)$ poszukiwanej sumy nie jest jednoznaczna, jednak jeśli przyjmiemy $t = n$, to wówczas jest ona jednoznaczna.

ZADANIE 2 – ŚLIMAK, ŚLIMAK, POKAŻ ROGI

Zadanie zaproponował: dr inż. Mariusz Pleszczyński, Wydział Matematyki Stosowanej, Politechnika Śląska

Napisz program, który dla zadanej liczby naturalnej n rysował będzie w konsoli ślimaka (przykładowe ślimaki przedstawione są na poniższym rysunku).

$n = 1$ $n = 3$



Liczba pętli muszli ślimaka wyznaczana jest przez wartość n, długość głowy ślimaka (część na prawo od muszli) jest równa połowie długości muszli (mierząc tę muszlę na górze). Czułki ślimaka mają wysokość równą połowie muszli licząc od górnej części głowy ślimaka. Oczy na czułkach są okręgami o promieniu proporcjonalnym do n (proporcjonalność tę można dobrać według własnego uznania). Oczy na głowie są okręgami, których środki wyznaczone są przez położenie czułek (współrzędne x) oraz $\frac{1}{3}$ wysokości głowy i mają promień stały (niezależny od n). Uśmiech ślimaka jest fragmentem paraboli, której wierzchołek leży po środku oczu (współrzędna x) i w $\frac{1}{3}$ wysokości głowy licząc od dołu tej głowy (współrzędna y), a współczynnik a tej paraboli dobrany jest tak, że uśmiech kończy się w połowie głowy (współrzędna y) natomiast ślimak uśmiecha się „od oka do oka” (współrzędna x).

Na powyższych rysunkach nie zachowane są proporcje, ogólnie, dla $n = 1$ muszla ślimaka (mierząc ją na górze) ma długość 2 (rysunek lewy), a dla $n = 3$ muszla ślimaka (mierząc ją na górze) ma długość 6 (rysunek prawy).

ZADANIE 3 – ZADANIE OSTRE JAK PIŁA, I WANT TO PLAY A GAME

Zadanie zaproponował: dr inż. Mariusz Pleszczyński, Wydział Matematyki Stosowanej, Politechnika Śląska

Pewna gra polega na tym, że pierwszy z uczestników (zakładamy, że gra w tę grę dwie osoby, chociaż równie dobrze może to być gra wieloosobowa) pisze na kartce wybraną przez siebie literę. Następnie gracz drugi dopisuje do tej litery, przed nią lub za nią, własną literę. Następnie tę samą czynność wykonuje gracz pierwszy, potem znowu drugi, itd. Gra kończy się, jeśli po wykonaniu ruch przez jednego z graczy powstanie wyraz znajdujący się w ustalonym przez graczy słowniku. W grze tej nie bierzemy pod uwagę słów jednoliterowych, więc gracz pierwszy może rozpocząć grę, pisząc np. literę *w* lub *u*.

Kolejną ważną zasadą w tej grze jest możliwość sprawdzenia, po dopisaniu litery przez przeciwnika, czy gracz ten nie blefuje, tzn. czy istnieje wyraz zawierający w sobie daną zbitkę liter. Przykładowo, jeżeli do zbitki liter *mys* (litery pojawiały się w kolejności: *s*, *ys*, *mys* – żadna z tych zbitek nie tworzy wyrazu, ale wszystkie te zbitki zawiera np. wyraz *mysz*) gracz dopisze z prawej strony *j*, tworząc zbitkę *mysj*, to przeciwnik może nabrać podejrzeń, czy istnieje wyraz zawierający tę zbitkę – wówczas gracz, który ją utworzył, musi podać przykład takiego wyrazu. Jeśli poda taki wyraz, to wygrywa, jeśli nie – przegrywa.

Istnienie wyrazu również musi być zasygnalizowane przez przeciwnika. Gdyby gracz rozpoczynający napisał literę *y*, jego przeciwnik dopisał z przodu literę *m*, tworząc *my*, to gracz pierwszy ma szansę na zwycięstwo, o ile zgłosi, że wyraz *my* znajduje się w słowniku. Jeśli to zgłosi – wygrywa, jeśli przeoczy ten fakt, to gra toczy się dalej. Jednak jeśli zgłosi, że dana zbitka tworzy wyraz, ale w słowniku nie będzie takiego wyrazu, to przegrywa.

W pliku *słownik.txt* znajduje się słownik, z którego korzystał będzie poniżej wspomniany program, w którym słowa (każde w nowej linii) posortowane są alfabetycznie.

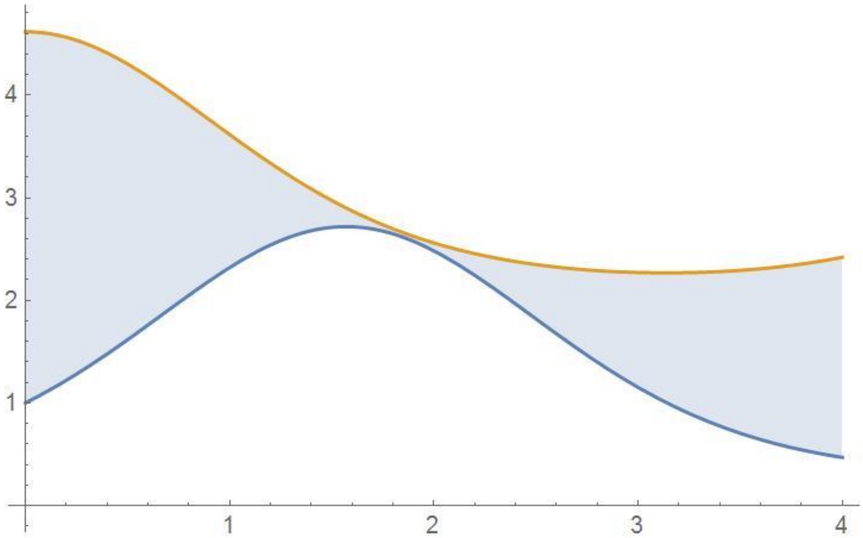
Napisz program, który rozgrywał będzie z człowiekiem partie takich gier. W każdej partii losowany jest rozpoczynający (komputer lub człowiek). Komputer ma dążyć do wygranej, im programista rozpatrzy więcej przypadków, tym lepiej (np. dopisywanie tylko takich liter, które tworzą zbitkę nie będącą wyrazem, ale dla której istnieje wyraz ją zawierający; reagowanie na powstanie wyrazu ze słownika i zgłaszanie tego; żądanie podania wyrazu, który zawiera daną zbitkę; w przypadku, gdy komputer nie może dopisać litery tak, żeby nie przegrać – czyli albo będzie utworzony wyraz, albo będzie zbitka, której żaden wyraz nie zawiera, to nastąpi próba blefu i komputer dopisze literę na jeden z dwóch przegrywających sposobów).

Przykładowa gra: *ę* → *ęd* → *węd* → *wędk* → gracz rozpoczynający przegrywa, bo dopisując z przodu literę naraża się na sprawdzenie blefu, a dopisując z prawej – albo powstanie wyraz, albo nastąpi sprawdzenie (o ile nie gracz drugi tego nie przeoczy).

ZADANIE 4 – STRZAŁY W POLU

Zadanie zaproponował: dr inż. Mariusz Pleszczyński, Wydział Matematyki Stosowanej, Politechnika Śląska

Załóżmy, że dane są dwie funkcje $f(x)$ i $g(x)$ określone dla każdego $x \in \langle a, b \rangle$ i takie, że w całym tym przedziale $f(x) \geq g(x)$. Zatem w tym przedziale te dwie krzywe ograniczają pewne pole, którego przykład przedstawiony jest na poniższym rysunku (funkcja f zaznaczona jest kolorem pomarańczowym, g – niebieskim, a pole szarym).



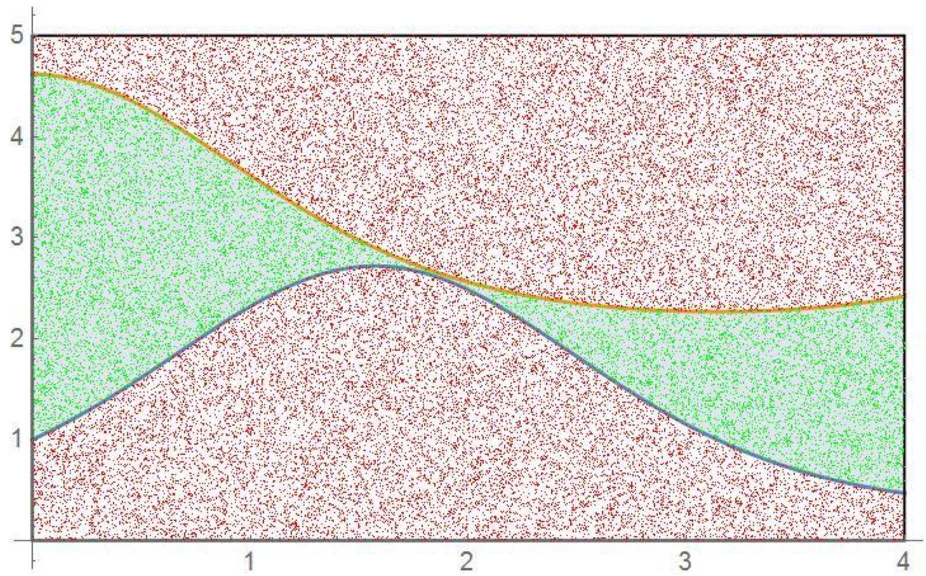
Chcemy obliczyć przybliżoną wartość tego pola. Jednym ze sposobów rozwiązania tego zadania jest skojarzenie tego problemu z prawdopodobieństwem geometrycznym – jeśli na rysunku tym wyobrazimy sobie tarczę będącą np. prostokątem o wierzchołkach w punktach $(0,0), (4,0), (4,5)$ i $(0,5)$, to oddając do tej tarczy losowy strzał, prawdopodobieństwo trafienia w obszar między krzywymi f i g wynosi tyle, ile wynosi stosunek poszukiwanego pola do pola prostokąta.

Oczywiście oddanie jednego losowego strzału nic nam nie powie o tym polu, ale już oddanie dużej liczby strzałów – tak, bo wtedy liczba strzałów, które trafiły w szary obszar do liczby wszystkich strzałów, ma się tak, jak ma się wartość poszukiwanego pola do wartości pola prostokąta.

Na poniższym rysunku przedstawiona jest sytuacja po oddaniu 50 000 losowych strzałów (na zielono oznaczone są strzały, które trafiły w szary obszar, a na czerwono te, które nie trafiły).

POLITECHNIKA ŚLĄSKA

Wydział Matematyki Stosowanej
Studenckie Koło Naukowo-Informatyczne
„Link”
ul. Kaszubska 23, 44-100 Gliwice



Napisz program, który dla zadanych współrzędnych prostokąta (wystarczą cztery liczby – a i b z dziedziny funkcji f i g oraz igrekowe współrzędne dwóch niesąsiadujących wierzchołków) i dla danej liczby wszystkich losowych strzałów (zakładamy, że każdy strzał trafia w tarczę), wyznaczał będzie przybliżoną wartość pola wyznaczonego przez krzywe f i g (igrekowe współrzędne prostokąta są tak dobrane, aby dla $x \in \langle a, b \rangle$ pole to zawierało się w tym prostokącie).

Rozwijając omówiony powyżej pomysł (niech S_p oznacza pole prostokąta, t – liczbę losowych strzałów, które trafiły w obszar, n – liczbę wszystkich oddanych losowych strzałów, a S – szukane pole), mamy:

$$\frac{S}{S_p} \approx \frac{t}{n} \quad \Rightarrow \quad S \approx \frac{t \cdot S_p}{n},$$

a ponieważ znamy wartości S_p, n oraz t (tę ostatnią oczywiście trzeba wyznaczyć w programie), to możemy w ten sposób wyznaczyć poszukiwaną przybliżoną wartość pola S.

Program przetestuj dla danych (przez D oznaczamy dokładną wartość pola):

a) $f(x) = 2{,}25 - x$, $g(x) = x(2 - x)$, $a = 0$, $b = 2$, $y_1 = 0$, $y_2 = 3$, $D = \frac{7}{6}$;

b) $f(x) = x + \cos x - \frac{1}{2}$, $g(x) = (x - 1)\sin x$, $a = 0$, $b = \pi$, $y_1 = -1$, $y_2 = 2$, $D = \frac{\pi^2 - 3\pi + 4}{2}$;

c) $f(x) = \sqrt{1 - x^2}$, $g(x) = -\sqrt{1 - x^2}$, $a = -1$, $b = 1$, $y_1 = -1$, $y_2 = 1$, $D = \pi$;

d) $f(x) = 2^{\cos x} + 1{,}15$, $g(x) = 2^{\sin x}$, $a = 0$, $b = 4$, $y_1 = 0$, $y_2 = 4$, $D \approx 2{,}9486907689$.

ALGORYTMION
Zespół „Algorytmion”
Politechnika Śląska
Wydział Matematyki Stosowanej
ul. Kaszubska 23
44-100 Gliwice

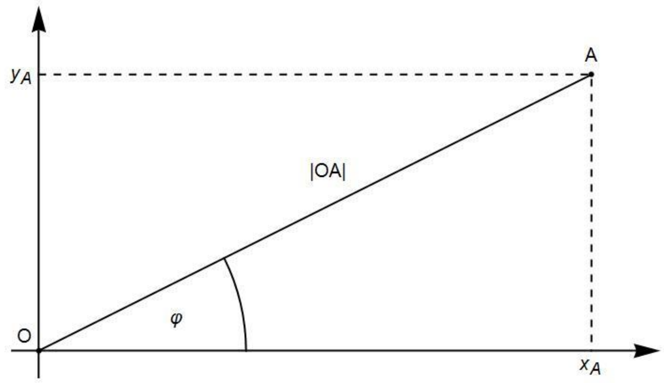
Studenckie Koło Naukowo-Informatyczne „Link”
Politechnika Śląska
Wydział Matematyki Stosowanej
ul. Kaszubska 23
44-100 Gliwice

ZADANIE 5 – BIEGUNOWO

Zadanie zaproponował: dr inż. Mariusz Pleszczyński, Wydział Matematyki Stosowanej, Politechnika Śląska

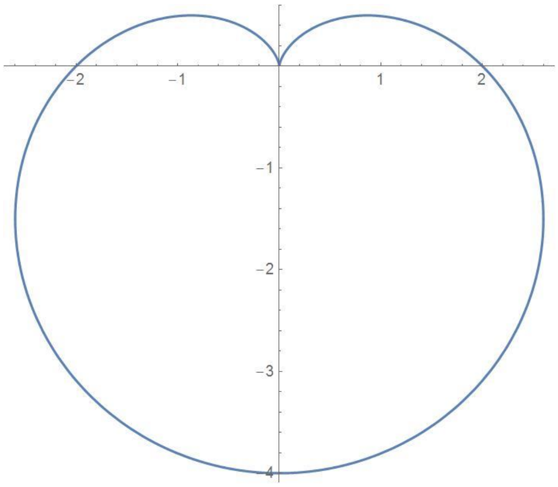
Dla dowolnego punktu A na płaszczyźnie (poza początkiem układu współrzędnych) możemy określić jednoznacznie kąt skierowany φ (jeśli przyjmiemy, że kąt ten jest z przedziału $\langle 0,2\pi)$), który tworzy dodatnia półoś iksów z półprostą wychodzącą z początku układu współrzędnych i przechodzącą przez punkt A. Przykładowo dla punktu $A=(1,1)$ kąt ten ma miarę $45°$, a dla punktu $A=(-\sqrt{3},-1)$ miara ta wynosi $\frac{7\pi}{6}$ radianów.

Dzięki temu kątowi można położenie dowolnego punktu na płaszczyźnie określić nieco inaczej niż klasycznie – podając wspomniany powyżej kąt (czyli określając położenie półprostej wyznaczonej przez początek układu współrzędnych i przez określany przez nas punkt) i odległość pomiędzy punktem $(0,0)$ i danym punktem A.



Jeżeli teraz określimy pewną nieujemną funkcję $\varrho(\varphi)$, która dla dowolnego kąta $\varphi \in \langle \varphi_1, \varphi_1 \rangle$ wyznacza odległość punktu A od początku układu współrzędnych, to przebiegając kolejne wartości φ, powstające w ten sposób punkty wykreślą pewną krzywą na płaszczyźnie. Przykładowo, jeśli $\varrho(\varphi)$ jest funkcją stałą, $\varphi \in \langle 0,2\pi)$, to mamy wówczas równanie okręgu o środku w punkcie $(0,0)$ i promieniu równym tej stałej (zobacz również zadanie 3 z 2017 roku).

Interesować nas będzie długość krzywej, jaką utworzą punkty $\left(\varphi, \varrho(\varphi)\right)$ (pamiętaj, że punkty te nie są zadane jako współrzędne punktu w kartezjańskim układzie współrzędnych). Przykładowo, dla $\varrho(\varphi)=2(1-\sin\varphi)$, $\varphi \in \langle 0,2\pi)$, interesować nas będzie obwód „serduszka” przedstawionego na poniższym rysunku.



Długość taką możemy wyznaczyć jako sumę długości odcinków powstałych z podziału przedziału kątów $\langle\varphi_1, \varphi_2\rangle$ przy pomocy n równoodległych punktów (kątów), z których pierwszy jest równy φ_1, a ostatni (n-ty) φ_2.

Napisz program, który na podstawie argumentów φ_1, φ_2 i n oznaczających odpowiednio kąt początkowy, kąt końcowy i gęstość podziału, obliczał będzie przybliżoną wartość długości krzywej powstałej z funkcji $\varrho(\varphi)$ dla $\varphi \in \langle\varphi_1, \varphi_2\rangle$.

Program przetestować możesz na danych (oznaczmy przez L wartość dokładną, a przez l – przybliżoną):

a) $\varrho(\varphi) = 2$, $\varphi \in \langle\frac{\pi}{2}, \pi\rangle$, $n = 200$, $l = 3{,}14158$, $L = \pi$ (ćwiartka okręgu o promieniu równym 2);
b) $\varrho(\varphi) = 2(1 - \sin\varphi)$, $\varphi \in \langle\frac{\pi}{2}, \frac{3\pi}{2}\rangle$, $n = 300$, $l = 7{,}999918$, $L = 8$ (lewa połowa „serduszka”);
c) $\varrho(\varphi) = \sin 4\varphi$, $\varphi \in \langle 0, \frac{\pi}{4}\rangle$, $n = 50$, $l = 2{,}1441$, $L \approx 2{,}14461$;
d) $\varrho(\varphi) = \pi^{\varphi}$, $\varphi \in \langle 0{,}2\pi\rangle$, $n = 500$, $l = 1763{,}9907$, $L \approx 1764.0024$.

Ponieważ zadanie to może sprawić kłopoty rozwiązującym, część punktów za to zadanie można uzyskać pisząc program rozwiązujący problem z pierwszego akapitu tego zadania – dla podanego punktu na płaszczyźnie program ma zwracać wartość kąta skierowanego $\varphi \in \langle 0{,}2\pi)$. Kąt ten ma być podany w stopniach i w radianach.