MINUT 2022 (4), s. 37–46
ISSN 2719-3063

Donata DĘBICKA

V Liceum Ogólnokształcące im. Klaudyny Potockiej w Poznaniu,
ul. Zmartwychwstańców 10, 61-501 Poznań

O rozwiązywaniu zadań konkursu matematyczno-informatycznego KOALA

Streszczenie. W artykule prezentowany jest drużynowy konkurs matematyczno-informatyczny KOALA dla młodzieży z Wielkopolski, którego celem jest rozbudzanie i rozwijanie zainteresowania matematyką i informatyką, w szczególności takimi dziedzinami jak: kombinatoryka, algorytmika i logika (stąd nazwa konkursu: KOALA). Ponadto celem konkursu jest doskonalenie umiejętności pracy w grupie, dyskusji oraz prezentowania własnych rozwiązań zadań i uzasadniania ich poprawności. W roku szkolnym 2021/2022 odbyła się już IX edycja konkursu. W artykule przedstawione zostaną metody (strategie) przydatne podczas rozwiązywania niektórych typów zadań konkursowych.

Słowa kluczowe: konkurs, kombinatoryka, algorytmika, logika.

1. Wstęp

W dzisiejszych czasach na rynku pracy jest bardzo duże zapotrzebowanie na osoby o wysoko rozwiniętych umiejętnościach myślenia w zakresie dziedzin, które tworzą podwaliny współczesnej informatyki (myślenie kombinatoryczne, myślenie algorytmiczne, myślenie logiczne).

W 2013 roku w V Liceum Ogólnokształcącym im. Klaudyny Potockiej w Poznaniu narodził się pomysł pozytywnego działania na rzecz dowartościowania kształcenia informatycznego w szkołach na terenie Wielkopolski w postaci opracowania i zorganizowania konkursu z pogranicza matematyki i informatyki, skierowanego wtedy do uczniów szkół gimnazjalnych.

Od roku szkolnego 2014/2015 konkurs KOALA organizowany jest w dwóch kategoriach wiekowych. W szczególności, po uwzględnieniu zmian w systemie oświaty od roku 2019/2020 mamy dwie jego edycje:

- dla uczniów klas 7 i 8 szkół podstawowych,
- dla uczniów szkół ponadpodstawowych.

Od tego samego roku konkurs KOALA odbywa się też na Dolnym Śląsku [5], a zwycięzcy z obu regionów każdego roku rywalizują w meczu międzyregionalnym Wielkopolska-Dolny Śląsk.

Autor korespondencyjny: D. Dębicka (ddebicka@vlo.poznan.pl).
Data wpłynięcia: 14.04.2022.

Od 2015 roku organizatorem konkursu obok V Liceum Ogólnokształcącego jest Zakład Matematyki Dyskretnej na Wydziale Matematyki i Informatyki Uniwersytetu im. Adama Mickiewicza.
Od lat konkurs wspierają też merytorycznie i organizacyjnie inne osoby (m.in. wykładowca Wydziału Informatyki i Telekomunikacji Politechniki Poznańskiej) oraz Poznański Oddział Polskiego Towarzystwa Matematycznego.

Konkurs Koala ma na celu rozbudzanie i rozwijanie zainteresowań młodzieży matematyką i informatyką. Zadania konkursowe mają zachęcać do poszukiwania oryginalnych i efektywnych rozwiązań. Dodatkowo celem konkursu jest również doskonalenie umiejętności pracy w grupie, dyskusji oraz prezentowania własnych rozwiązań zadań i uzasadniania ich poprawności.

Czytelnik niniejszego czasopisma z pewnością nie ma wątpliwości, iż myślenie matematyczne polega na czymś więcej niż na rachunkach. O roli rozwiązywania zadań w kształtowaniu myślenia matematycznego wypowiadało się wielu wybitnych matematyków i dydaktyków, m.in. George Pólya [3], Andrzej Góralski [1], John Mason [2].

Staramy się, aby na zestawy zadań naszego konkursu składały się ciekawe zadania z kombinatoryki, algorytmiki i logiki – dziedzin, które tworzą podwaliny współczesnej informatyki. Wiele z nich jest zadaniami autorskimi, opracowanymi przez organizatorów konkursu. Na stronie internetowej konkursu [4] znaleźć można większość zadań konkursowych. W przygotowaniu do konkursu mogą pomóc uczniom także zasoby dydaktyczne w stylu Computer Science Unplugged [6], [7].

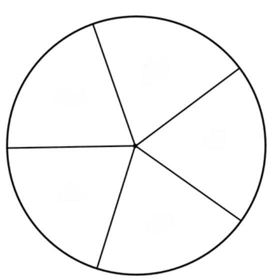
Od kilku lat finalistom konkursu Koala przysługują punkty brane pod uwagę w postępowaniu rekrutacyjnym do szkół średnich, gdy jako szczególne osiągnięcia są wymienione na świadectwie ukończenia szkoły podstawowej.

W dalszej części artykułu przedstawię metody (strategie) przydatne podczas rozwiązywania niektórych zadań konkursowych. Posłużę się zadaniami *Turniej strzelecki* oraz *Aleja eukaliptusowa* z finału 8. edycji konkursu w wersji dla szkół podstawowych.

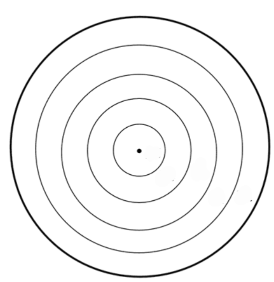
2. Przykład zadania z kombinatoryki – Turniej strzelecki

Treść zadania:

W pewnym turnieju strzeleckim tarcza została podzielona na pięć sektorów (Rysunek 1) i pięć pierścieni (Rysunek 2), co ostatecznie daje 25 pól.

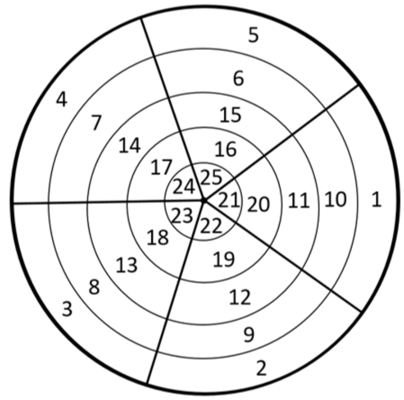


Rysunek 1.



Rysunek 2.

Do każdego pola została przypisana liczba punktów, które zawodnik dostaje, gdy w to pole trafi (Rysunek 3).



Rysunek 3.

Zasady turnieju są następujące:

– Każdy zawodnik oddaje serię 5 strzałów i za każdym razem musi trafić w tarczę.

– W każdy sektor może trafić tylko raz.

– W każdy pierścień może trafić tylko raz.

Naruszanie którejkolwiek z powyższych zasad skutkuje przyznaniem zawodnikowi 0 punków za całą serię.

– Za trafienie w sam środek tarczy lub w linię pomiędzy polami przyznaje się 0 punktów (za strzał).

– Wygrywa zawodnik z największą liczbą punktów, czyli największą sumą punktów uzyskanych za trafienia w poszczególne pola.

Ile najwięcej punktów może uzyskać zwycięzca? Ile jest różnych wyborów pól przez zawodnika, dających tę liczbę punktów?

Rozwiązanie:

Jak widać na Rysunku 3, w pierścieniu nr 1 (najbardziej zewnętrzny) znajdują się liczby od 1 do 5, w pierścieniu 2 – od 6 do 10, w pierścieniu 3 – od 11 do 15 itd.

Przez a_{ij} oznaczymy liczbę w pierścieniu o numerze i oraz sektorze o numerze j, gdzie $i, j = 1, 2, \ldots, 5$.
Jeśli i jest liczbą nieparzystą, to możemy zapisać: $a_{ij} = 5(i-1) + j$.
Jeśli i jest liczbą parzystą, to możemy zapisać: $a_{ij} = 5(i-1) + (6-j)$.

Dla dowolnej permutacji j_1, j_2, j_3, j_4, j_5 liczb 1, 2, 3, 4, 5 otrzymujemy:
$a_{1j_1} + a_{2j_2} + a_{3j_3} + \ldots + a_{5j_5} = j_1 + [5 + (6 - j_2)] + (5 \cdot 2 + j_3) + [5 \cdot 3 + (6 - j_4)] + (5 \cdot 4 + j_5) = 5 \cdot (1 + 2 + 3 + 4) + 2 \cdot 6 + (j1 + j3 + j5) - (j2 + j4) = 5 \cdot 10 + 12 + (j_1 + j_3 + j_5) - (j_2 + j_4) = 50 + 12 + (j_1 + j_3 + j_5) - (j_2 + j_4) = 62 + (j_1 + j_3 + j_5) - (j_2 + j_4)$.

Suma ta ma największą wartość, gdy $j_2 + j_4$ jest liczbą możliwie najmniejszą, to znaczy, gdy np. $j_2 = 1$, $j_4 = 2$.

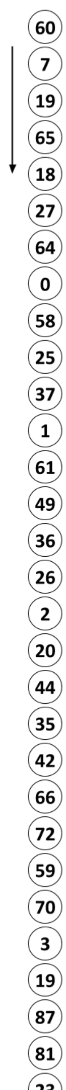
Wtedy maksymalna suma to $62 + (3 + 4 + 5) - (1 + 2) = 62 + 12 - 3 = 71$.

Ile jest różnych wyborów pięciu liczb, dających tę największą sumę?

– Liczba permutacji w zbiorze j_1, j_3, j_5 jest równa 6 (pola: 5, 14, 23; 5, 13, 24; 4, 15, 23; 4, 13, 25; 3, 14, 25; 3, 15, 24).
– Liczba permutacji w zbiorze j_2, j_4 jest równa 2 (pola: 10, 19; 9, 20).

Największą sumę można otrzymać z $6 \cdot 2 = 12$ różnych wyborów 5 pól na tarczy.

3. Przykład zadania z algorytmiki – Aleja eukaliptusowa



Treść zadania:

Miś koala przemieszcza się z drzewa na drzewo w poszukiwaniu drzewa o coraz większej liczbie liści eukaliptusowych. Zakładamy, że drzewa są rozmieszczone jak na rysunku, a koala nie cofa się, czyli wędruje na południe (jak to pokazuje strzałka). Celem koali jest odwiedzenie jak największej liczby drzew z pysznymi liśćmi.

Które drzewo koala powinien wybrać jako pierwsze? Które drzewa powinien wybierać podczas wędrówki? Odpowiedź uzasadnij.

Rozwiązanie:

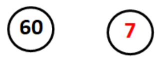
Najpierw ustalmy, o co w zadaniu chodzi, co mamy wyznaczyć, określmy wszystkie warunki (są napisane, ale musimy przetłumaczyć to na „nasze”).

Poszukiwanie drzew o coraz większej liczbie liści powinniśmy zinterpretować matematycznie jako zadanie polegające na wybraniu z ciągu liczb (zapisanych na rysunku jedna pod drugą) takich liczb, które tworzą podciąg rosnący (oczywiście bez zmiany ich początkowej kolejności). Celem koali jest odwiedzenie jak największej liczby drzew, a więc naszym celem jest znalezienie podciągu nie tylko rosnącego, ale przy tym najdłuższego.

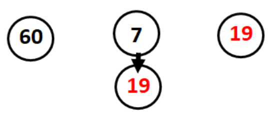
Nietrudno zauważyć, że podciągów rosnących dla ciągu liczb z naszego zadania jest bardzo dużo. Skąd będzie wiadomo, że jakiś wybrany podciąg rzeczywiście będzie tym najdłuższym? W tym momencie musimy dobrze przemyśleć i zaplanować metodę poszukiwania tego podciągu i dopiero potem będziemy w stanie ustalić rozwiązanie.

Od czego zacząć?
Spróbujmy zilustrować graficznie wszystkie możliwe podciągi rosnące, które da się zbudować z liczb zapisanych na kilku początkowych drzewach. Zobaczmy, jak mógłby wyglądać przebieg rozwiązywania zadania, krok po kroku.

Krok 1. Wybierzmy drzewo z 60 liśćmi. Przyjmijmy, że liczba 60 rozpoczyna jakiś podciąg rosnący.



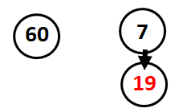
Krok 2. Spójrzmy na drzewo z siedmioma liśćmi. Liczby 7 nie dołączymy do liczby 60, bo $7 < 60$. Liczba 7 może natomiast rozpoczynać inny podciąg rosnący. Mamy zatem w tym momencie dwa podciągi jednoelementowe.

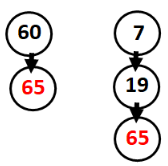


Krok 3. Pojawia się kolejne drzewo, tym razem z 19 liśćmi. Liczby 19 nie dołączymy do 60, ale do 7 dołączyć możemy i powstanie podciąg (7, 19). Mamy też trzecią możliwość: 19 jako pierwszy element innego podciągu.

Ale czy ma sens brać pod uwagę nowy podciąg rozpoczynający się od 19, skoro wcześniej na drzewach wystąpiły już liczby mniejsze od 19 i liczba 19 może być drugim elementem wcześniej rozpoczętego podciągu?

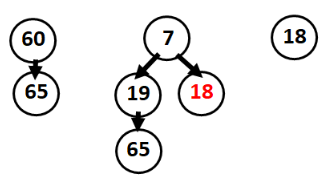
Ponieważ szukamy podciągu najdłuższego, zrezygnujemy z rozpoczynania kolejnego podciągu, o ile mamy możliwość przedłużenia ciągu już istniejącego. W konsekwencji zostaniemy przy następujących dwóch podciągach: jednoelementowym i dwuelementowym (jak na rysunku z prawej).





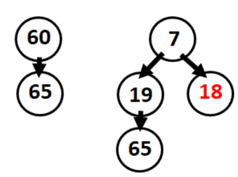
Krok 4. Dla liczby 65 z kolejnego drzewa, biorąc pod uwagę ustalenia odnotowane w kroku 3, można utworzyć dwa podciągi (jak na rysunku z lewej).

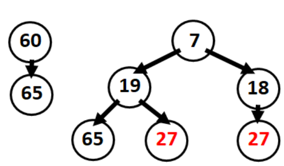
Liczba 65 mogłaby być także drugim elementem podciągu postaci (7, 65), ale tę propozycję odrzucimy, ponieważ korzystniejsze jest, aby koala wybrał trójelementowy drugi podciąg.



Krok 5. Tym razem patrzymy na drzewo z 18 liśćmi. Liczba 18 może rozpoczynać nowy podciąg. Może też być dołączona do ciągu rozpoczętego liczbą 7 w kroku 2, ale wtedy liczby 19 i 65 nie mogą wchodzić w skład takiego podciągu, czyli tworzymy podciąg (7, 18).

Analogicznie, jak w kroku 3, zrezygnujemy z tworzenia podciągu rozpoczynającego się liczbą 18, ponieważ ta liczba może być ulokowana na drugiej pozycji w podciągu (7, 18). Zostajemy przy układzie podciągów jak na rysunku z prawej. Te układy będą punktem wyjścia do znajdowania podciągów dłuższych.





Krok 6. A gdzie umieścić liczbę z kolejnego drzewa, które ma 27 liści? Po 19, po 18, a może rozpocząć nowy podciąg?

Gdybyśmy dalej postępowali w podobny sposób, jak ten opisany w krokach od 1 do 5, to w końcu otrzymalibyśmy rozwiązanie. Ale długa droga przed nami… i łatwo się pomylić.

Spróbujmy nieco uprościć zapis kolejnych etapów rozwiązywania zadania i jednocześnie przedstawimy przykład rozumowania, które zawiera przekonujące uzasadnienie poprawności rozwiązania.
Posłużymy się tabelką, w której w sposób systematyczny będziemy gromadzić informacje o podciągach rosnących, które da się utworzyć po rozpatrzeniu kolejnych liczb ciągu.

Dla pięciu początkowych elementów ciągu, czyli podciągu (60, 7, 19, 65 i 18), zgromadzone informacje przedstawimy tak:

Ciąg liczb wyrażający liczbę liści na drzewie	Numer ***n*** elementu w ciągu liczb	Numer ***m*** elementu w podciągu rosnącym. Aktualnie rozpatrywany element ciągu wpisany na odpowiedniej pozycji ***m***.										Długość najdłuższego rosnącego podciągu, kończącego się elementem ciągu o numerze ***n***
		1	2	3	4	5	6	7	8	9	10	
60	1	60										1
7	2	7										1
19	3		19									2
65	4			65								3
18	5		18									2

W kolumnie oznaczonej daną liczbą m przechowujemy informację o pozycji, na której znajduje się rozpatrywana w danym momencie liczba oryginalnego ciągu (wskazana w wierszu opisanym daną liczbą n) w najdłuższym do tej pory znalezionym podciągu rosnącym, w którym ta liczba występuje.

Opiszemy teraz metodę umieszczania kolejnych elementów w tabeli. Przypuśćmy, że umieściliśmy w niej już $n-1$ elementów ciągu i chcemy umieścić w niej n-ty wyraz. Ten wyraz umieścimy w kolumnie o numerze m, jeśli spełnione są dwa warunki:

$W1$: w kolumnie o numerze $m-1$ znajduje się co najmniej jeden element ciągu mniejszy od wyrazu n-tego.

$W2$: wyraz n-ty jest mniejszy bądź równy od wszystkich wyrazów znajdujących się dotychczas w kolumnie m-tej lub w tej kolumnie nie ma na razie żadnych liczb.

Jeśli żadna z kolumn o numerach $m >= 2$ nie spełnia obu warunków, to wyraz n-ty umieszczamy w kolumnie $m = 1$.

Można zauważyć, że:

– spełnienie warunku $W1$ zapewnia przedłużenie podciągu rosnącego.

– spełnienie warunku $W2$ gwarantuje, że kolejne elementy podciągów rosnących będą znajdowały się w kolejnych kolumnach tabeli, przez co łatwo określimy długość najdłuższego rosnącego podciągu, kończącego się n-tym elementem ciągu.

Przeanalizujmy proces umieszczania w tabeli szóstego elementu ciągu.

Którym elementem w możliwie najdłuższym podciągu rosnącym będzie 27? Wyznaczmy zatem m.
Czy m może być równe 4? Nie, ponieważ warunek $W1$ nie jest spełniony (w kolumnie o numerze 3 nie ma ani jednego elementu mniejszego od elementu 27).
Czy m może być równe 3? Tak, ponieważ warunki $W1$ i $W2$ są spełnione (w kolumnie o numerze 2 znajdują się elementy 19 i 18, które są mniejsze od 27 i jednocześnie liczba 27 jest mniejsza od wszystkich wyrazów znajdujących się obecnie w kolumnie numer 3).

Po wpisaniu szóstego elementu ciągu na odpowiedniej pozycji, tabela przyjmuje następującą postać:

Ciąg liczb wyrażający liczbę liści na drzewie	Numer ***n*** elementu w ciągu liczb	Numer ***m*** elementu w podciągu rosnącym. Aktualnie rozpatrywany element ciągu wpisany na odpowiedniej pozycji ***m***.										Długość najdłuższego rosnącego podciągu, kończącego się elementem ciągu o numerze ***n***
		1	2	3	4	5	6	7	8	9	10	
60	1	60										1
7	2	7										1
19	3		19									2
65	4			65								3
18	5		18									2
27	6			27								3

Rozpatrzmy siódmy element ciągu.

Którym elementem w możliwie najdłuższym podciągu rosnącym będzie 64? Dla rozpatrywanego wyrazu wyznaczmy m.

Czy m może być równe 4? Tym razem tak.

Warunek $W1$ jest spełniony – w kolumnie o numerze 3 znajduje się element 27, który jest mniejszy od rozpatrywanego elementu 64.

Warunek $W2$ także jest spełniony – w kolumnie numer 4 nie ma żadnych liczb.

Po umieszczeniu siódmego elementu ciągu na odpowiedniej pozycji w tabeli, przyjmuje ona następującą postać:

Ciąg liczb wyrażający liczbę liści na drzewie	Numer ***n*** elementu w ciągu liczb	Numer ***m*** elementu w podciągu rosnącym. Aktualnie rozpatrywany element ciągu wpisany na odpowiedniej pozycji ***m***.										Długość najdłuższego rosnącego podciągu, kończącego się elementem ciągu o numerze ***n***
		1	2	3	4	5	6	7	8	9	10	
60	1	60										1
7	2	7										1
19	3		19									2
65	4			65								3
18	5		18									2
27	6			27								3
64	7				64							4

Dalej kontynuujemy wypełnianie tabeli. Po kolei dla każdego kolejnego (aż do trzydziestego) n-tego elementu ciągu wyznaczymy liczbę m, która informować będzie nas, którym elementem możliwie najdłuższego podciągu rosnącego (utworzonego z elementów ciągu od pierwszego do n-tego) może być ten n-ty element ciągu.

Efekt pracy przedstawiamy poniżej:

Ciąg liczb wyrażający liczbę liści na drzewie	Numer ***n*** elementu w ciągu liczb	Numer ***m*** elementu w podciągu rosnącym. Aktualnie rozpatrywany element ciągu wpisany na odpowiedniej pozycji ***m***.										Długość najdłuższego rosnącego podciągu, kończącego się elementem ciągu o numerze ***n***
		1	2	3	4	5	6	7	8	9	10	
60	1	60										1
7	2	7										1
19	3		19									2
65	4			65								3
18	5		18									2
27	6			27								3
64	7				64							4
0	8	0										1
58	9				58							4
25	10			25								3
37	11				37							4
1	12		1									2
61	13					61						5
49	14					49						5
36	15				36							4
26	16				26							4
2	17			2								3
20	18				20							4
44	19					44						5
35	20					35						5
42	21						42					6
66	22							66				7
72	23								72			8
59	24							59				7
70	25								70			8
3	26				3							4
19	27					19						5
87	28									87		9
81	29									81		9
23	30						23					6

Warto dopowiedzieć, że opisana wyżej strategia rozwiązywania zadań jest nazywana czasami programowaniem dynamicznym.

Odczytujemy z tabelki, że długość najdłuższego podciągu rosnącego w naszym zadaniu wynosi 9, gdyż największe m, które otrzymaliśmy w trakcie wypełniania tabeli to 9.

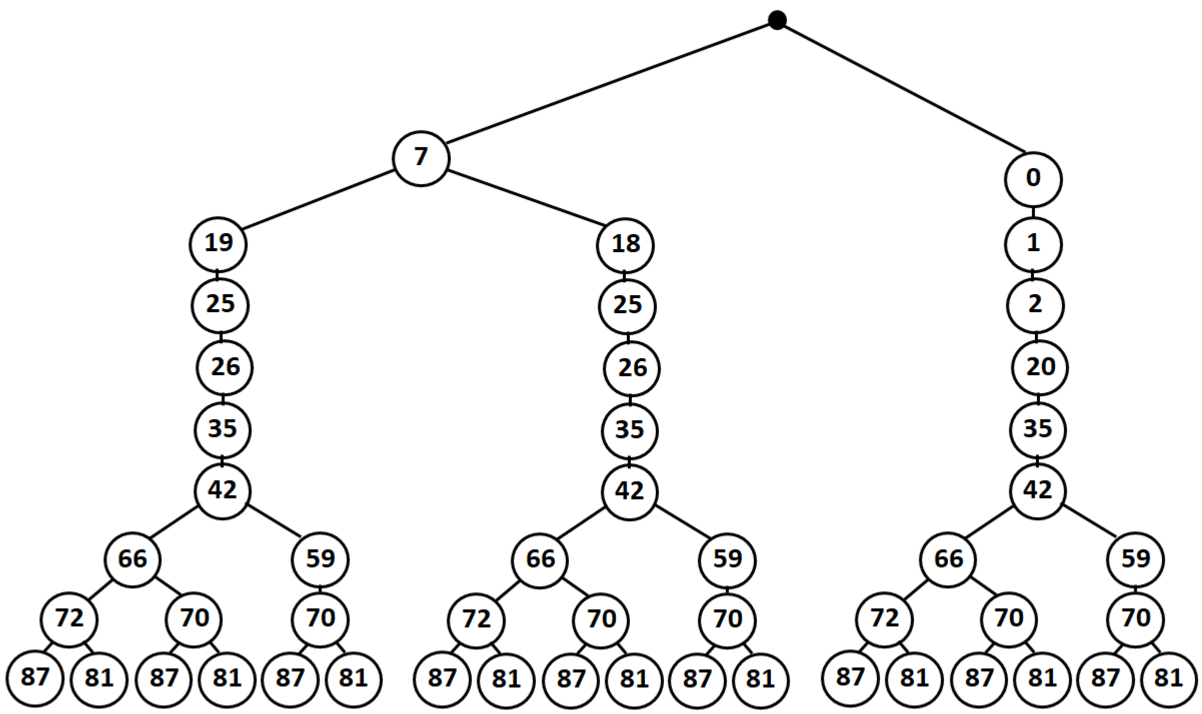
Podciąg rosnący o maksymalnej długości możemy wyznaczyć, zaczynając od ostatniego wyrazu (znajdującego się w kolumnie $m = 9$) i przechodząc tabelę w górę w lewo (jak po schodkach). W każdym kroku wyznaczania szukanego podciągu tą metodą należy wybierać elementy coraz mniejsze o coraz mniejszych numerach n z kolumn kolejno o numerach $9, 8, 7, \ldots 1$, nie pomijając żadnej.

Które z drzew koala powinien wybrać jako pierwsze? Odpowiedź brzmi: drugie lub ósme. Dlaczego pierwsze drzewo nie? Ponieważ w drugiej kolumnie nie ma liczby, która byłaby większa od 60 i mniejsza od 65.

Które drzewa powinien wybierać podczas kolejnych dni wędrówki? Jest wiele rozwiązań, przykładowy wybór zaznaczono kolorem szarym w tabeli: $(7, 18, 25, 26, 35, 42, 59, 70, 87)$.

Zadanie zostało w pełni rozwiązane.

Możemy dodatkowo narysować diagram przedstawiający wszystkie możliwe podciągi rosnące długości 9.



Uwaga: Ograniczenie się do wskazania przykładowego podciągu rosnącego o długości 9 nie jest pełnym rozwiązaniem zadania. Jest to tylko uzasadnienie, że istnieje co najmniej jeden podciąg rosnący o długości co najmniej 9.

4. Ocenianie rozwiązań

Przy ocenianiu rozwiązań zadań (w II etapie konkursu oraz w finale) stosujemy ocenianie holistyczne (całościowe), rozszerzone o pewne elementy charakterystyczne dla naszego konkursu (przyznawane są dodatkowe punkty za rozwiązanie efektywne, poprawne pod względem językowym).

Dlaczego ocenianie holistyczne? Takie podejście zakłada przyjęcie poziomów wykonania zadania. Oceniane są: pokonanie zasadniczej trudności zadania oraz kolejne czynności prowadzące do pełnego rozwiązania zadania. Dopuszczamy różne dobre, poprawne rozwiązania, także takie, których nie przewidzieliśmy. Interesuje nas, w jaki sposób, poprawny pod względem metodologicznym, uczeń doszedł do swoich wniosków.

Literatura

1. A. Góralski, *Twórcze rozwiązywanie zadań*, Państwowe Wydawnictwo Naukowe, Warszawa 1980.
2. J. Mason, L. Burton, K. Stacey, *Matematyczne myślenie*, Wydawnictwa Szkolne i Pedagogiczne, Warszawa 2005.
3. G. Polya, *Jak to rozwiązać?* , Wydanie III, Wydawnictwo Naukowe PWN, Warszawa 2011.
4. https://koala.poznan.pl [dostęp 11.05.2022].
5. http://www.math.uni.wroc.pl/fmw/dla-uczniów/koala/konkurs-koala-kombinatoryka-algorytmikalogika [dostęp 11.05.2022].
6. https://bezkomputera.wmi.amu.edu.pl/ppi/index.html [dostęp 11.05.2022].
7. http://jasijoasia.edu.pl/ [dostęp 11.05.2022].