Rozwiązania testu wiedzy algorytmicznej

XX OIJ, zawody I stopnia
29 października 2025

Olimpiada Informatyczna Juniorów

1. **Rozważmy poniższą funkcję:**

wersja C++

```
int f(int x)
{
    if (x%2 == 0)
        return 0;
    else
        return x;
}
```

wersja Python

```
def f(x):
    if x%2 == 0:
        return 0
    else:
        return x
```

Jaki będzie wynik działania `f(11)+f(12)+f(13)`**?**

24

Rozwiązanie: Funkcja przyjmuje liczbę x. Jeśli jest ona parzysta, funkcja zwraca wynik 0, a jeśli nieparzysta, zwraca wynik x. Zatem `f(11) = 11`, `f(12) = 0`, `f(13) = 13`, a odpowiedź to $11 + 0 + 13 = 24$.

2. **Rozważmy poniższy fragment kodu:**

wersja C++

```
int f(string s)
{
    int wynik = 0;
    for (char c : s)
    {
        if (c == 'x')
            wynik += 1;
        if (c == 'y')
            wynik += 2;
    }
    return wynik;
}
```

wersja Python

```
def f(s):
    wynik = 0
    for c in s:
        if c == 'x':
            wynik += 1
        if c == 'y':
            wynik += 2
    return wynik
```

Jaki będzie wynik wywołania `f("xyyzxyzy")`**?**

10

Rozwiązanie: Funkcja przegląda kolejne znaki napisu podanego jako parametr (od lewej do prawej). Za każdy `x` do zmiennej `wynik` dodawane jest 1, a za każdy `y` dodawane jest 2. Dla innych znaków nic się nie dzieje. W napisie są 2 znaki `x` oraz 4 znaki `y`, stąd wynik to 10.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

3. **Rozpatrzmy następujący fragment kodu:**

wersja C++

```
a = a + b + c;
b = a - b - c;
c = a - b - c;
a = a - b - c;
```

wersja Python

```
a = a + b + c
b = a - b - c
c = a - b - c
a = a - b - c
```

Jakie były początkowe wartości a**,** b**,** c**, jeżeli po wykonaniu powyższych linii kodu mamy** a=4**,** b=3**,** c=7**? Wypisz je kolejno oddzielone przecinkami, np.** $1, 2, 3$**.**

3, 7, 4

Rozwiązanie: Niech na początku w zmiennej a była wartość A, w zmiennej b wartość B, a w zmiennej c wartość C. Po pierwszej instrukcji w zmiennych a, b, c będą kolejno wartości $A + B + C, B, C$, po drugiej instrukcji $A + B + C, A, C$, po trzeciej $A + B + C, A, B$ i wreszcie po czwartej $-$ C, A, B. W takim razie $A = 3$, $B = 7$, $C = 4$.

4. **Zsumowano wszystkie liczby naturalne od** 1 **do** 100 **włącznie, oprócz jednej, i otrzymano wynik** 5000**. Jaka liczba została pominięta?**

50

Rozwiązanie: Aby obliczyć sumę $1 + 2 + \ldots + 100$, można połączyć w pary 1 i 100, 2 i 99, 3 i 98, $\ldots$, 50 i 51 itd. Wszystkie sumy są równe 101, jest ich 50, więc cała suma wynosi 5050. Zatem brakuje liczby 50.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

5. **Trzy z czterech poniższych funkcji są równoważne (robią to samo). Które to są? (Zaznacz odpowiednie kratki na karcie odpowiedzi.)**

☒ **A.** *wersja C++*

```
int f(int a, int b)
{
    return (a > b) ? a : b;
}
```

wersja Python

```
def f(a, b):
    return a if a > b else b
```

☒ **B.** *wersja C++*

```
int f(int a, int b)
{
    if (a >= b)
        return a;
    else
        return b;
}
```

wersja Python

```
def f(a, b):
    if a >= b:
        return a
    else:
        return b
```

☒ **C.** *wersja C++*

```
int f(int a, int b)
{
    if (b > a)
        return b;
    return a;
}
```

wersja Python

```
def f(a, b):
    if b > a:
        return b
    return a
```

☐ **D.** *wersja C++*

```
int f(int a, int b)
{
    if (a > b)
        return a;
    else if (b > a)
        return b;
    else
        return 0;
}
```

wersja Python

```
def f(a, b):
    if a > b:
        return a
    elif b > a:
        return b
    else:
        return 0
```

Rozwiązanie: Trzy pierwsze kody wypisują zawsze większą z liczb a, b – zwróć uwagę, że jeśli liczby są równe, to można wypisać którąkolwiek. Ostatni wypisuje 0, jeśli liczby są równe.

6. **Ile jest ciągów znaków o długości 7 złożonych z liter `A`, `B` i `C` takich, że żadne dwie kolejne litery nie są takie same?**

192

Rozwiązanie: Jeśli zaczniemy od `A`, to następną literą musi być `B` lub `C`. Jeśli wybierzemy np. `B`, to znowu są dwie możliwości na kolejną literę (`A`, `C`), i tak dalej. Tak więc pierwszą literę wybieramy na 3 sposoby, a każdą kolejną na 2. Łącznie możliwości jest zatem $3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 = 3 \cdot 2^6 = 192$.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

7. **Rozważmy poniższy fragment kodu:**

wersja C++

```
int f(int n)
{
    int res = 0;
    int w = 1;
    while (n > 0)
    {
        int d = n % 4;
        n /= 4;
        res += d * w;
        w *= 3;
    }
    return res;
}
```

wersja Python

```
def f(n):
    res = 0
    w = 1
    while n > 0:
        d = n % 4
        n //= 4
        res += d * w
        w *= 3
    return res
```

Jaki będzie wynik wywołania `f(50)`?

29

Rozwiązanie: Każdy krok algorytmu oblicza resztę z dzielenia n przez 4, a potem dzieli samą liczbę n przez 4. Dla $n = 50$ kolejne reszty będą wynosiły 2, 0, 3. (Można zauważyć, że jest to zapis n w systemie czwórkowym). Z kolei zmienna w będzie w kolejnych iteracjach pętli przybierać wartości równe kolejnym potęgom trójki: $1, 3, 9, 27, \ldots$. Wartości te mnożone są przez kolejne reszty i sumowane, zatem wynik to $2 \cdot 1 + 0 \cdot 3 + 3 \cdot 3^2 = 29$. (Tu również jest pewne podobieństwo do zapisu w systemie trójkowym.)

8. **Ze zbioru $\{20, 8, 3, 2, 1\}$ wybieramy niektóre liczby (co najmniej jedną, być może wszystkie, żadnej liczby nie można wybrać dwukrotnie) i sumujemy. Ile różnych sum możemy w ten sposób uzyskać?**

27

Rozwiązanie: Widać, że z liczb $1, 2, 3$ można uzyskać dokładnie 6 różnych sum. Dołączenie liczby 8 podwaja możliwą liczbę sum ($1, 2, \ldots, 6$ oraz $8+1, 8+2, \ldots, 8+6$) i dodaje jeszcze jedną sumę (8) – dzieje się tak, ponieważ $8 > 6$ i sumy się nie powtarzają. Zatem po dorzuceniu 8 mamy 13 możliwych sum. Podobnie, dorzucenie teraz 20 podwaja liczbę sum i dodaje jeszcze jedną, więc wynik to $2 \cdot 13 + 1 = 27$.

9. **Ile gwiazdek wypisze poniższy kod?**

wersja C++

```
int x = 100;
int y = x;

while (y > 0)
{
    for(int i = 0 ; i < x; i++)
        cout << "*" << "\n";
    y = y / 2;
}
```

wersja Python

```
x = 100
y = x

while y > 0:
    for i in range(x):
        print("*")
    y = y // 2
```

700

Rozwiązanie: Ponieważ cały czas `x = 100`, więc każda iteracja wewnętrznej pętli wypisuje 100 gwiazdek. Z kolei zmienna `y` będzie równa kolejno $100, 50, 25, 12, 6, 3, 1$, a po siódmej iteracji mamy `y = 0` i następnej już nie bedzie. W tych siedmiu iteracjach łącznie wypisze się 700 gwiazdek.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

10. **Jakie wartości mogą przyjąć zmienne X i Y, aby poniższy fragment programu wypisał TAK?**

wersja C++

```
if( X && !Y )
    cout << "TAK" << "\n";
```

wersja Python

```
if X and not Y:
    print("TAK")
```

☐ **A.** `X = 1, Y = 1`

☒ **B.** `X = 5, Y = 0`

☐ **C.** `X = 0, Y = -1`

☐ **D.** `X = 'n', Y = '0'`

Rozwiązanie: Zarówno w C++, jak i Pythonie, liczby są zamieniane na wartość „prawda" lub „fałsz" w ten sposób, że 0 jest „fałszem", a każda inna liczba „prawdą". Z kolei pojedyncze znaki to zawsze „prawda".

11. **Po wykonaniu poniższszego fragmentu programu, ile razy na wyjściu pojawi się liczba 3?**

wersja C++

```
int n = 4;
for(int i = 1; i < n; i++)
    for(int j = 2; j < n; j++)
        for(int k = 3; k < n; k++)
           cout << i << " " << j << " "
                << k << "\n";
```

wersja Python

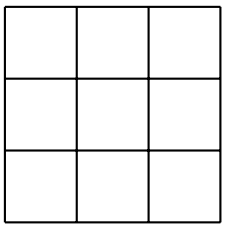
```
n = 4
for i in range(1, n):
    for j in range(2, n):
        for k in range(3, n):
            print(i,j,k)
```

11

Rozwiązanie: Trójka może wypisać się jako wartość `i`, jako wartość `j` lub `k`. Jako wartość `i` wypisze się $2 \cdot 1 = 2$ razy, ponieważ dla `i = 3` pozostałe zmienne przebiegają wszystkie swoje wartości (`j` od 2 do 3 a `k` od 3 do 3). Ale podobnie jest z trójką jako wartością `j`: wypisze się dla `i =` $1, 2, 3$ oraz `k =` 3, a więc $3 \cdot 1 = 3$ razy. Podobnie trójka jako wartość `k` wypisze się $3 \cdot 2 = 6$ razy, więc łączny wynik to 11.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

12. **Dana jest tabela złożona z trzech wierszy i trzech kolumn, jak na rysunku. W komórkach tabeli należy rozmieść cyfry od 1 do 9, w taki sposób, żeby suma cyfr zapisanych w każdym wierszu, w każdej kolumnie i dla każdej przekątnej wynosiła 15. Cyfry nie mogą się powtarzać.**
Istnieje kilka poprawnych rozwiązań, ale w każdym z nich jedna z cyfr nie zmienia swojego położenia. Jaka to cyfra?



5

Rozwiązanie: Przykładowe rozłożenie cyfr:

4	3	8
9	5	1
2	7	6

Jeżeli obrócimy tabelkę o $90°$, dostaniemy inne poprawne rozwiązanie, w którym tylko liczba 5 pozostanie na swoim miejscu. Zatem 5 musi być poszukiwaną odpowiedzią.
Trikowe-matematyczne rozwiązanie: Suma obu przekątnych oraz środkowego wiersza i środkowej kolumny musi zawsze wynosić $4 \cdot 15 = 60$. Ale w tej sumie każdy element tabeli występuje raz, z wyjątkiem środkowego, który jest policzony 4 razy. Zatem na wartość 60 składa się suma wszystkich liczb $(1 + 2 + \ldots + 9 = 45)$ oraz jeszcze trzykrotnie powtórzony środkowy element, który musi wobec tego zawsze wynosić 5.

13. **Rozważmy poniższą funkcję:**

wersja C++

```
int f(int a, int k)
{
    if (k == 0)
        return a;
    else
        return f(a / k, k - 1);
}
```

wersja Python

```
def f(a, k):
    if k == 0:
        return a
    else:
        return f(a // k, k - 1)
```

Jaka jest najmniejsza liczba całkowita dodatnia `a` taka, że `f(a,4)` jest równe 2?

48

Rozwiązanie: Ponieważ `f(a,k) = f(a/k, k-1)`, mamy `f(a,4) = f(a/4, 3) = f(a/12, 2) = f(a/24, 1) = f(a/24, 0) = a/24`. Najmniejsza liczba, dla której `a/24 = 2`, to 48.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

14. **Rozważmy poniższą funkcję:**

wersja C++

```
void f(int p, int q)
{
    if (p > q)
        return;
    int s = (p + q) / 2;
    f(s+1, q);
    cout << s << "\n";
    f(p, s-1);
}
```

wersja Python

```
def f(p, q):
    if p > q:
        return
    s = (p + q) // 2
    f(s+1, q)
    print(s)
    f(p, s-1)
```

Wywołano `f(2000,3000)`**. Jaka będzie czwarta z kolei wypisana liczba?**

2997

Rozwiązanie: Najlepiej sprawdzić na kilku małych przykładach, co tak naprawdę robi funkcja `f` – okazuje się, że wywołanie `f(p,q)` po prostu wypisuje wszystkie liczby między p a q włącznie, od największej do najmniejszej. Zatem w tym wypadku wypisane zostaną kolejno $3000, 2999, 2998, 2997$, i tak dalej.

15. **Ile jest liczb całkowitych dodatnich, których suma cyfr w zapisie dziesiętnym wynosi dokładnie** 10**, a iloczyn cyfr jest równy przynajmniej** 30**?**

23

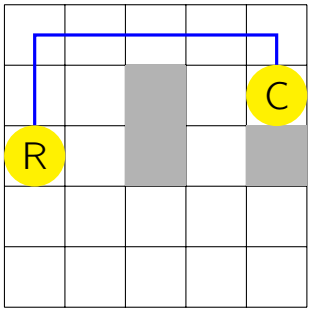
Rozwiązanie: Gdyby choć jedną cyfrą tej liczby było 1, nie moglibyśmy osiągnąć iloczynu 30 (największy możliwy iloczyn to $1 \cdot 3 \cdot 3 \cdot 3 = 27$). Rozważmy, ile razy może wystąpić w rozkładzie cyfra 2:

- Jeśli dwójek jest 4 lub więcej, to musi to być liczba 22222.
- Jeśli są dokładnie 3 dwójki, to ostatnią cyfrą musi być 4, co daje 4 możliwe liczby (4222, 2422, 2242, 2224).
- Jeśli są dokładnie 2 dwójki, to pozostałą sumę 6 trzeba rozłożyć na sumę, która nie zawiera jedynek ani dwójek. Jest tylko jeden taki rozkład: $6 = 3 + 3$, co daje 6 możliwych liczb: 2233, 2323, ... Pozostawienie 6 w całości (np. 226) daje zbyt mały iloczyn 24.
- Jeśli jest dokładnie 1 dwójka, to pozostałą sumę 8 trzeba rozłożyć tak, aby nie było w niej jedynek ani dwójek. Są tylko trzy możliwe rozkłady: $4 + 4$, $5 + 3$ i po prostu 8. Pierwszy generuje 3 możliwe liczby, drugi 6, trzeci ma za mały iloczyn ($2 \cdot 8 = 16$).
- Jeśli nie ma w ogóle dwójek, to liczba musi mieć co najmniej trzy cyfry (dwucyfrowe o sumie cyfr 10 mają za mały iloczyn). Są tylko trzy takie liczby: 334, 343, 433.

Łącznie mamy $1 + 4 + 6 + 3 + 6 + 3 = 23$ poszukiwane liczby.

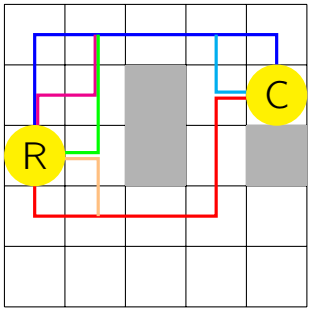
Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

16. **Poniższy schemat przedstawia początkową pozycję robota (R) na planszy wraz z oznaczoną najkrótszą drogą do celu (C). Robot może przemieszczać się jedynie na sąsiednie pola (góra/dół/prawo/lewo), nie może przemieszczać się na skos, ani nie może przemieszczać się po szarych polach. Ile takich najkrótszych dróg do celu dla tego robota istnieje?**



8

Rozwiązanie:



Wszystkie najkrótsze drogi wymagają 7 ruchów robota. Droga wzdłuż górnej krawędzi planszy (niebieska) ma 6 „wariantów": robot może wybrać w pierwszej części drogi wariant „niebieski", „fioletowy" lub „zielony", a w drugiej części drogi wariant „niebieski" lub „błękitny". Robot może też wybrać niższą drogę (czerwoną), która ma dwa warianty („czerwony" i "pomarańczowy"). Łącznie daje to 8 najkrótszych dróg.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

17. **Rozważmy poniższą funkcję:**

wersja C++

```
int f(string s)
{
    int wynik = 0;

    for (char c : s)
    {
        if (c != '+' && c != '-' && c != 'o')
            return -1;
        wynik = wynik * 3;
        if (c == '+')
            wynik++;
        if (c == '-')
            wynik--;
    }

    return wynik;
}
```

wersja Python

```
def f(s):
    wynik = 0

    for c in s:
        if c != '+' and c != '-' and c != 'o':
            return -1
        wynik = wynik * 3
        if c == '+':
            wynik += 1
        if c == '-':
            wynik -= 1

    return wynik
```

Podaj najkrótszy możliwy ciąg znaków `s` **składający się ze znaków** `+`**,** `o` **i** `-`**, dla którego** `f(s)` $= 156$**.**

`+-o-+o`

Rozwiązanie: W każdym okrążeniu pętli zmienna `wynik` mnoży się przez 3, a następnie dodawane jest do niej -1, 0 lub $+1$, w zależności od napotkanego znaku. Skoro ostateczny rezultat (156) jest podzielny przez 3, to ostatnim znakiem musiało być `o`, a przed tym okrążeniem w zmiennej `wynik` musiało znajdować się $156/3 = 52$. Z kolei przedostatnim znakiem musiał być `+`, bo liczba 52 nie jest podzielna przez 3, za to 51 jest. Zatem 52 musiało zostać otrzymane z 51 przez dodanie 1, a przedtem w zmiennej `wynik` znajdowało się $51/3 = 17$. Trzecim od końca znakiem musiał być `-`, i tak dalej – postępując w ten sposób możemy odtworzyć całą sekwencję znaków.
Zadanie nawiązuje do nietypowego systemu zapisu liczb, tzw. *systemu trójkowego zrównoważonego* (Wikipedia), oczywiście nie trzeba go jednak znać, by rozwiązać zadanie.

18. **Grupa uczniów próbuje ustawić się według wzrostu od najmniejszego do największego. Stoją w rzędzie, a wartości ich wzrostu w centymetrach to kolejno:** $163, 161, 168, 164, 162, 166, 165, 169, 167$**. W jednym ruchu jeden z uczniów może wyjść ze swojego miejsca i wejść w dowolnie wybrane inne miejsce. Ile minimalnie ruchów trzeba wykonać, żeby uczniowie stanęli w kolejności rosnącej?**

5

Rozwiązanie: Zastanówmy się, ilu uczniów może się **nie** poruszyć. Tacy uczniowie muszą od początku stać w kolejności rosnącej. Wobec tego maksymalnie możemy pozostawić na miejscach czworo uczniów (np. $161, 162, 165, 167$), bo nie można wybrać większej podgrupy, która stałaby w dobrej kolejności. Pozostałych pięcioro musi więc zostać przestawionych, i to jest minimalna liczba.

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".

19. **Rozważmy poniższą funkcję:**

wersja C++

```
int f(string s)
{
    int x = 0;
    for (char c : s)
    {
        if (x == 2 && c == 'b')
            x = 1;
        else if (x == 1 && c == 'b')
            x = 2;
        else if (x == 0 && c == 'a')
            x = 1;
    }
    if (x == 1)
        return true;
    else
        return false;
}
```

wersja Python

```
def f(s):
    x = 0
    for c in s:
        if x == 2 and c == 'b':
            x = 1
        elif x == 1 and c == 'b':
            x = 2
        elif x == 0 and c == 'a':
            x = 1
    if x == 1:
        return True
    else:
        return False
```

Zaznacz wszystkie ciągi znaków, dla których funkcja zwróci `true`/`True`**.**

☒ **A.** `acbcab`

☐ **B.** `abbbc`

☐ **C.** `bbbbbbc`

☒ **D.** `bcca`

Rozwiązanie: Wynik działania funkcji zależy od zmiennej `x`. Tak długo, jak w napisie nie ma znaku `a`, zmienna ta będzie stale równa 0. Po napotkaniu znaku `a`, zmienna `x` zmienia się z 1 na 2 i odwrotnie przy każdym znaku `b`. Zatem po literze `a` musi być parzysta liczba liter `b` – napisy **A** i **D** pasują do tego schematu, zaś **B** i **C** nie pasują.

20. **Specjalne słowa powstają przez doklejenie do słowa jego odwrotności (np.** `wyrazzaryw`**), a następnie zastąpienie niektórych liter gwiazdkami. Które z poniższych słów z pewnością nie mogą być specjalnymi słowami powstałymi w taki sposób?**

☒ **A.** `*ab****ab*`

☐ **B.** `a*a***b*`

☒ **C.** `m***d*m`

☐ **D.** `*a***a**`

Rozwiązanie: Specjalne słowo musi być palindromem, czyli czytany od przodu i tyłu musi być taki sam. Musi mieć również parzystą liczbę znaków, skoro powstaje przez sklejenie dwóch słów równej długości. W napisie **A** nie zgadza się druga litera (`a`) z przedostatnią (`b`), z kolei napis **C** ma nieparzystą liczbę znaków. Napisy **B** i **D** łatwo otrzymać metodą podaną zadaniu (np. ze słów, odpowiednio, `abak` i `naan`).

Olimpiada finansowana jest ze środków Ministerstwa Edukacji Narodowej w ramach zadania publicznego „Organizacja i przeprowadzenie olimpiad i turniejów w latach szkolnych 2025/2026, 2026/2027, 2027/2028".