1.Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego sprawdzającego, które z 99 wprowadzonych liczb są podzielne przez 5.

1. Blok startowy: START
2. Blok operacyjny: Inicjalizacja zmiennych i=0, licznik=0
3. Blok wprowadzenia: Wprowadź liczbę n (wprowadź 99 liczb)
4. Blok operacyjny: i = i + 1
5. Blok warunkowy: Sprawdź, czy n % 5 == 0 a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 8
6. Blok operacyjny: licznik = licznik + 1
7. Blok wypisania: Wypisz n (podzielna przez 5)
8. Blok warunkowy: Sprawdź, czy i < 99 a. Jeśli TAK: Przejdź do bloku 3 b. Jeśli NIE: Przejdź do bloku 9
9. Blok wypisania: Wypisz licznik (ilość liczb podzielnych przez 5)
10. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

2. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego sprawdzającego, czy wprowadzona z klawiatury liczba jest liczbą pierwszą.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę n
3. Blok operacyjny: Inicjalizacja zmiennych i=2, pierwsza=true
4. Blok warunkowy: Sprawdź, czy n <= 1 a. Jeśli TAK: Przejdź do bloku 10 b. Jeśli NIE: Przejdź do bloku 5
5. Blok warunkowy: Sprawdź, czy i \* i <= n a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 9
6. Blok warunkowy: Sprawdź, czy n % i == 0 a. Jeśli TAK: Przejdź do bloku 7 b. Jeśli NIE: Przejdź do bloku 8
7. Blok operacyjny: pierwsza = false
8. Blok operacyjny: i = i + 1, Przejdź do bloku 5
9. Blok warunkowy: Sprawdź, czy pierwsza == true a. Jeśli TAK: Wypisz "Liczba pierwsza", Przejdź do bloku 11 b. Jeśli NIE: Wypisz "Liczba złożona", Przejdź do bloku 11
10. Blok wypisania: Wypisz "Liczba nie jest pierwsza"
11. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

3. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczającego iloczyn 2*4*6\*...\*2n dla dowolnie ustalonego n ∈ N, podanego przez użytkownika.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę n
3. Blok operacyjny: Inicjalizacja zmiennych i=1, iloczyn=1
4. Blok operacyjny: i = i + 1
5. Blok operacyjny: iloczyn = iloczyn \* (2 \* i)
6. Blok warunkowy: Sprawdź, czy i < n a. Jeśli TAK: Przejdź do bloku 4 b. Jeśli NIE: Przejdź do bloku 7
7. Blok wypisania: Wypisz iloczyn
8. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

4. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego, który po wprowadzeniu liczby n wypisze sumę liczb parzystych mniejszych od n.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę n
3. Blok operacyjny: Inicjalizacja zmiennych i=2, suma=0
4. Blok warunkowy: Sprawdź, czy i < n a. Jeśli TAK: Przejdź do bloku 5 b. Jeśli NIE: Przejdź do bloku 8
5. Blok operacyjny: suma = suma + i
6. Blok operacyjny: i = i + 2
7. Przejdź do bloku 4
8. Blok wypisania: Wypisz suma
9. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

5. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego rozwiązującego równanie ax + b = 1 po podaniu przez użytkownika liczb a i b.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę a
3. Blok wprowadzenia: Wprowadź liczbę b
4. Blok warunkowy: Sprawdź, czy a == 0 a. Jeśli TAK: Przejdź do bloku 5 b. Jeśli NIE: Przejdź do bloku 6
5. Blok warunkowy: Sprawdź, czy b == 1 a. Jeśli TAK: Wypisz "Nieskończenie wiele rozwiązań", Przejdź do bloku 8 b. Jeśli NIE: Wypisz "Brak rozwiązań", Przejdź do bloku 8
6. Blok operacyjny: x = (1 - b) / a
7. Blok wypisania: Wypisz x (rozwiązanie równania)
8. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

6. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczający wartość a^b dla wprowadzonych niezerowych liczb całkowitych/naturalnych a i b.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę a
3. Blok wprowadzenia: Wprowadź liczbę b
4. Blok operacyjny: Inicjalizacja zmiennych wynik=1, i=1
5. Blok warunkowy: Sprawdź, czy i <= b a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 8
6. Blok operacyjny: wynik = wynik \* a
7. Blok operacyjny: i = i + 1, Przejdź do bloku 5
8. Blok wypisania: Wypisz wynik (wartość a^b)
9. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

7.Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego, który pozwoli na wprowadzenie 5 liczb oraz wypisze sumę wprowadzonych liczb dodatnich i ich ilość.

1. Blok startowy: START
2. Blok operacyjny: Inicjalizacja zmiennych suma=0, ilość=0, i=1
3. Blok warunkowy: Sprawdź, czy i <= 5 a. Jeśli TAK: Przejdź do bloku 4 b. Jeśli NIE: Przejdź do bloku 8
4. Blok wprowadzenia: Wprowadź liczbę liczba\_i
5. Blok warunkowy: Sprawdź, czy liczba\_i > 0 a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 7
6. Blok operacyjny: suma = suma + liczba\_i, ilość = ilość + 1
7. Blok operacyjny: i = i + 1, Przejdź do bloku 3
8. Blok wypisania: Wypisz "Suma liczb dodatnich: ", suma
9. Blok wypisania: Wypisz "Ilość liczb dodatnich: ", ilość
10. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

8.Macie w pdf gotowe.

9.Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego, który wyznaczy n początkowych wyrazów ciągu określonego następująco: a1 = 2, ak+1 = 1 + 2ak dla k ≥ 2.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę n
3. Blok operacyjny: Inicjalizacja zmiennych ak=2, k=1
4. Blok wypisania: Wypisz ak
5. Blok warunkowy: Sprawdź, czy k < n a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 9
6. Blok operacyjny: ak = 1 + 2 \* ak
7. Blok operacyjny: k = k + 1
8. Przejdź do bloku 4
9. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

10.Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego rozwiązującego równanie ax^2 + bx = 1 po podaniu przez użytkownika liczb a i b.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę a
3. Blok wprowadzenia: Wprowadź liczbę b
4. Blok warunkowy: Sprawdź, czy a == 0 a. Jeśli TAK: Przejdź do bloku 5 b. Jeśli NIE: Przejdź do bloku 6
5. Blok warunkowy: Sprawdź, czy b == 0 a. Jeśli TAK: Wypisz "Brak rozwiązań", Przejdź do bloku 11 b. Jeśli NIE: Przejdź do bloku 10
6. Blok operacyjny: Obliczenie Δ (delta) = b^2 - 4a(-1)
7. Blok warunkowy: Sprawdź, czy Δ < 0 a. Jeśli TAK: Wypisz "Brak rozwiązań", Przejdź do bloku 11 b. Jeśli NIE: Przejdź do bloku 8
8. Blok operacyjny: Obliczenie x1 = (-b - sqrt(Δ)) / (2a)
9. Blok operacyjny: Obliczenie x2 = (-b + sqrt(Δ)) / (2a)
10. Blok wypisania: Wypisz "Rozwiązania równania: x1 =", x1, "x2 =", x2
11. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

11.Macie w pdf

12.Macie w pdf.

13.Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego sumującego podawane przez użytkownika liczby do momentu podania przez niego liczby 0, a następnie wypisujący otrzymaną sumę i ilość podanych liczb (włącznie z zerem/niezerowych).

1. Blok startowy: START
2. Blok operacyjny: Inicjalizacja zmiennych suma=0, ilość=0
3. Blok wprowadzenia: Wprowadź liczbę liczba
4. Blok operacyjny: suma = suma + liczba
5. Blok warunkowy: Sprawdź, czy liczba != 0 a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 8
6. Blok operacyjny: ilość = ilość + 1
7. Przejdź do bloku 3
8. Blok wypisania: Wypisz "Suma podanych liczb: ", suma
9. Blok wypisania: Wypisz "Ilość podanych liczb niezerowych: ", ilość
10. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

14. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wypisującego wszystkie dzielniki podanej przez użytkownika liczby.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę liczba
3. Blok operacyjny: Inicjalizacja zmiennej i=1
4. Blok warunkowy: Sprawdź, czy i <= liczba a. Jeśli TAK: Przejdź do bloku 5 b. Jeśli NIE: Przejdź do bloku 8
5. Blok warunkowy: Sprawdź, czy liczba % i == 0 a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 7
6. Blok wypisania: Wypisz i
7. Blok operacyjny: i = i + 1, Przejdź do bloku 4
8. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

15. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczający najmniejszy wspólny dzielnik (NWD) dwóch liczb, używając algorytmu Euklidesa.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę a
3. Blok wprowadzenia: Wprowadź liczbę b
4. Blok warunkowy: Sprawdź, czy b == 0 a. Jeśli TAK: Przejdź do bloku 7 b. Jeśli NIE: Przejdź do bloku 5
5. Blok operacyjny: Obliczenie reszta = a % b
6. Blok operacyjny: Przypisanie a = b, b = reszta, Przejdź do bloku 4
7. Blok wypisania: Wypisz "Najmniejszy wspólny dzielnik: ", a
8. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

16.Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczający największy wspólny dzielnik (NWD) dwóch liczb, używając algorytmu Euklidesa.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę a
3. Blok wprowadzenia: Wprowadź liczbę b
4. Blok warunkowy: Sprawdź, czy b == 0 a. Jeśli TAK: Przejdź do bloku 7 b. Jeśli NIE: Przejdź do bloku 5
5. Blok operacyjny: Obliczenie reszta = a % b
6. Blok operacyjny: Przypisanie a = b, b = reszta, Przejdź do bloku 4
7. Blok wypisania: Wypisz "Największy wspólny dzielnik: ", a
8. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

17. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczający najmniejszą wspólną wielokrotność (NWW) dwóch liczb, korzystając z faktu, że NWW(a, b) = |a \* b| / NWD(a, b).

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę a
3. Blok wprowadzenia: Wprowadź liczbę b
4. Blok operacyjny: Inicjalizacja zmiennych a\_copy = a, b\_copy = b
5. Blok warunkowy: Sprawdź, czy b\_copy == 0 a. Jeśli TAK: Przejdź do bloku 8 b. Jeśli NIE: Przejdź do bloku 6
6. Blok operacyjny: Obliczenie reszta = a\_copy % b\_copy
7. Blok operacyjny: Przypisanie a\_copy = b\_copy, b\_copy = reszta, Przejdź do bloku 5
8. Blok operacyjny: Obliczenie NWW = abs(a \* b) / a\_copy
9. Blok wypisania: Wypisz "Najmniejsza wspólna wielokrotność: ", NWW
10. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

18. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczający wszystkie czynniki pierwsze podanej liczby.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę n
3. Blok operacyjny: Inicjalizacja zmiennej dzielnik = 2
4. Blok warunkowy: Sprawdź, czy dzielnik \* dzielnik <= n a. Jeśli TAK: Przejdź do bloku 5 b. Jeśli NIE: Przejdź do bloku 9
5. Blok warunkowy: Sprawdź, czy n % dzielnik == 0 a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 7
6. Blok wypisania: Wypisz dzielnik Blok operacyjny: n = n / dzielnik, Przejdź do bloku 4
7. Blok operacyjny: dzielnik = dzielnik + 1, Przejdź do bloku 4
8. Blok wypisania: Wypisz n (ostatni czynnik pierwszy)
9. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

19. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego znajdujący pierwszych 20 par liczb bliźniaczych.

1. Blok startowy: START
2. Blok operacyjny: Inicjalizacja zmiennych: p = 3, q = 5, liczba\_par = 0
3. Blok warunkowy: Sprawdź, czy liczba\_par < 20 a. Jeśli TAK: Przejdź do bloku 4 b. Jeśli NIE: Przejdź do bloku 10
4. Blok operacyjny: Inicjalizacja zmiennych: liczba = p, jest\_pierwsza = True
5. Blok warunkowy: Sprawdź, czy liczba <= p / 2 a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 8
6. Blok warunkowy: Sprawdź, czy p % liczba == 0 a. Jeśli TAK: Przejdź do bloku 7 b. Jeśli NIE: Przejdź do bloku 11
7. Blok operacyjny: jest\_pierwsza = False, Przejdź do bloku 8
8. Blok warunkowy: Sprawdź, czy jest\_pierwsza == True a. Jeśli TAK: Przejdź do bloku 9 b. Jeśli NIE: Przejdź do bloku 12
9. Blok wypisania: Wypisz p, q Blok operacyjny: liczba\_par = liczba\_par + 1, Przejdź do bloku 12
10. Blok stopowy: STOP
11. Blok operacyjny: liczba = liczba + 1, Przejdź do bloku 5
12. Blok operacyjny: p = p + 2, q = q + 2, Przejdź do bloku 3

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

20. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego znajdujący pięć najmniejszych liczb naturalnych 𝑛, takich, że liczba 𝑛^2 − 1 jest iloczynem trzech różnych liczb pierwszych.

1. Blok startowy: START
2. Blok operacyjny: Inicjalizacja zmiennych: n = 2, znalezione\_liczby = 0
3. Blok warunkowy: Sprawdź, czy znalezione\_liczby < 5 a. Jeśli TAK: Przejdź do bloku 4 b. Jeśli NIE: Przejdź do bloku 11
4. Blok operacyjny: oblicz x = n^2 - 1
5. Blok operacyjny: Inicjalizacja zmiennych: dzielnik = 2, liczba\_czynnikow\_pierwszych = 0
6. Blok warunkowy: Sprawdź, czy dzielnik \* dzielnik <= x a. Jeśli TAK: Przejdź do bloku 7 b. Jeśli NIE: Przejdź do bloku 10
7. Blok warunkowy: Sprawdź, czy x % dzielnik == 0 a. Jeśli TAK: Przejdź do bloku 8 b. Jeśli NIE: Przejdź do bloku 9
8. Blok operacyjny: x = x / dzielnik, liczba\_czynnikow\_pierwszych = liczba\_czynnikow\_pierwszych + 1 Przejdź do bloku 6
9. Blok operacyjny: dzielnik = dzielnik + 1, Przejdź do bloku 6
10. Blok warunkowy: Sprawdź, czy liczba\_czynnikow\_pierwszych == 3 a. Jeśli TAK: Przejdź do bloku 12 b. Jeśli NIE: Przejdź do bloku 13
11. Blok stopowy: STOP
12. Blok wypisania: Wypisz n Blok operacyjny: znalezione\_liczby = znalezione\_liczby + 1, Przejdź do bloku 13
13. Blok operacyjny: n = n + 1, Przejdź do bloku 3

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

21. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego sprawdzający, czy podane dwie liczby są pierwsze.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę a
3. Blok wprowadzenia: Wprowadź liczbę b
4. Blok operacyjny: Inicjalizacja zmiennych: liczba = 2, a\_jest\_pierwsza = True, b\_jest\_pierwsza = True
5. Blok warunkowy: Sprawdź, czy liczba <= a / 2 lub liczba <= b / 2 a. Jeśli TAK: Przejdź do bloku 6 b. Jeśli NIE: Przejdź do bloku 11
6. Blok warunkowy: Sprawdź, czy a % liczba == 0 a. Jeśli TAK: Przejdź do bloku 7 b. Jeśli NIE: Przejdź do bloku 8
7. Blok operacyjny: a\_jest\_pierwsza = False
8. Blok warunkowy: Sprawdź, czy b % liczba == 0 a. Jeśli TAK: Przejdź do bloku 9 b. Jeśli NIE: Przejdź do bloku 10
9. Blok operacyjny: b\_jest\_pierwsza = False
10. Blok operacyjny: liczba = liczba + 1, Przejdź do bloku 5
11. Blok warunkowy: Sprawdź, czy a\_jest\_pierwsza == True a. Jeśli TAK: Przejdź do bloku 12 b. Jeśli NIE: Przejdź do bloku 13
12. Blok wypisania: Wypisz, że a jest liczbą pierwszą
13. Blok warunkowy: Sprawdź, czy b\_jest\_pierwsza == True a. Jeśli TAK: Przejdź do bloku 14 b. Jeśli NIE: Przejdź do bloku 15
14. Blok wypisania: Wypisz, że b jest liczbą pierwszą
15. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

22. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego liczącego sumę cyfr podanej liczby naturalnej.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź liczbę n
3. Blok operacyjny: Inicjalizacja zmiennych: suma\_cyfr = 0
4. Blok warunkowy: Sprawdź, czy n > 0 a. Jeśli TAK: Przejdź do bloku 5 b. Jeśli NIE: Przejdź do bloku 8
5. Blok operacyjny: Oblicz cyfra = n % 10
6. Blok operacyjny: suma\_cyfr = suma\_cyfr + cyfra
7. Blok operacyjny: n = n // 10, Przejdź do bloku 4
8. Blok wypisania: Wypisz sumę cyfr
9. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

23. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien typ operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczający wszystkie dwucyfrowe liczby Armstronga.

1. Blok startowy: START
2. Blok operacyjny: Inicjalizacja zmiennych: liczba = 10
3. Blok warunkowy: Sprawdź, czy liczba < 100 a. Jeśli TAK: Przejdź do bloku 4 b. Jeśli NIE: Przejdź do bloku 8
4. Blok operacyjny: Oblicz cyfra1 = liczba // 10
5. Blok operacyjny: Oblicz cyfra2 = liczba % 10
6. Blok operacyjny: Oblicz suma\_poteg = cyfra1**2 + cyfra2**2
7. Blok warunkowy: Sprawdź, czy suma\_poteg == liczba a. Jeśli TAK: Przejdź do bloku 9 b. Jeśli NIE: Przejdź do bloku 10
8. Blok stopowy: STOP
9. Blok wypisania: Wypisz liczbę Armstronga (liczba)
10. Blok operacyjny: liczba = liczba + 1, Przejdź do bloku 3

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

24. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien rodzaj operacji. Poniżej znajduje się opis algorytmu blokowego wyznaczający pole prostokąta o podanych bokach.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź długość boku a
3. Blok wprowadzenia: Wprowadź długość boku b
4. Blok operacyjny: Oblicz pole\_prostokata = a \* b
5. Blok wypisania: Wypisz pole\_prostokata
6. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.

25. Algorytm blokowy służy do wizualizacji kolejnych kroków algorytmu w postaci prostokątnych bloków, gdzie każdy z nich reprezentuje pewien rodzaj operacji. Poniżej znajduje się opis algorytmu blokowego sprawdzający, czy dwa okręgi o podanych środkach i promieniach mają punkty wspólne.

1. Blok startowy: START
2. Blok wprowadzenia: Wprowadź współrzędne środka okręgu 1: (x1, y1)
3. Blok wprowadzenia: Wprowadź promień okręgu 1: r1
4. Blok wprowadzenia: Wprowadź współrzędne środka okręgu 2: (x2, y2)
5. Blok wprowadzenia: Wprowadź promień okręgu 2: r2
6. Blok operacyjny: Oblicz odległość między środkami okręgów: d = sqrt((x2 - x1)^2 + (y2 - y1)^2)
7. Blok operacyjny: Oblicz sumę promieni: suma\_promieni = r1 + r2
8. Blok warunkowy: Sprawdź, czy d <= suma\_promieni i d >= abs(r1 - r2) a. Jeśli TAK: Przejdź do bloku 9 b. Jeśli NIE: Przejdź do bloku 10
9. Blok wypisania: Wypisz "Okręgi mają punkty wspólne"
10. Blok wypisania: Wypisz "Okręgi nie mają punktów wspólnych"
11. Blok stopowy: STOP

W algorytmie blokowym używa się strzałek, aby wskazać, do którego bloku należy przejść po zakończeniu danego etapu. W powyższym opisie kroki algorytmu zostały przedstawione jako lista, ale w rzeczywistości powinny być zorganizowane jako schemat bloków połączonych strzałkami, które wskazują kierunek przepływu danych i kontroli.